

APPENDIX

J:\Does\37791\00195\01204599.DOC

Recognition of Merged Characters Based on Forepart Prediction, Necessity-Sufficiency Matching, and Character-Adaptive Masking

Qiqiang Song, Member, IEEE, Zuo Li, Michael R. Lyu, Fellow, IEEE, and Shijie Cai

Abstract—Merged characters are the major cause of recognition errors. We classify the merging relationship between two involved characters into three types: “linear,” “nonlinear,” and “overlapped.” Most segmentation methods handle the first type well, however, their capabilities of handling the other two types are limited. The weakness of handling the nonlinear and overlapped types results from character segmentation by linear, usually vertical, cuts assumed in these methods. This paper proposes a novel merged character segmentation and recognition method based on forepart prediction, necessity-sufficiency matching and character-adaptive masking. This method utilizes the information obtained from the forepart of merged characters to predict candidates for the leftmost character, and then applies character-adaptive masking and character recognition to verifying the prediction. Therefore, the arbitrary-shaped cutting path will follow the right shape of the leftmost character so as to preserve the shape of the next character. This method handles the first two types well and greatly improves the segmentation accuracy of the overlapped type. The experimental results and the performance comparisons with other methods demonstrate the effectiveness of the proposed method.

Index Terms—Character-adaptive masking, character feature extraction, character segmentation, forepart prediction, merged character recognition, necessity-sufficiency matching.

I. INTRODUCTION

OPTICAL character recognition (OCR) is one of the most successful applications of automatic pattern recognition. Nowadays, the research interests of OCR focuses on the recognition of degraded, multiple-font machine-printed, and handwritten text [1]. For printed text, the main difficulty comes from the severely merged or degraded characters. Until now, incorrect recognition of merged characters is still one of the main causes for recognition errors. The well-known tests of commercial printed-text OCR systems by the University of Nevada, Las Vegas [2], show that even when perfect patterns were recognized, commercial systems experience 0.5% spacing errors in average. This is essentially a segmentation error by a process

that attempts to isolate word images [3]. Furthermore, since segmentation errors often raise chain effects, the performance of segmentation is crucial for the whole OCR process.

This paper proposes an effective merged-character recognition method, which consists of a novel segmentation scheme based on forepart prediction (FP), character-adaptive masking (CAM), and a necessity-sufficiency matching (NSM) algorithm for single character recognition. First, the method predicts the candidates for the foremost (for horizontally aligned text, “foremost” equals “leftmost”) character of merged characters based on a set of forepart features and then cuts it from the image using its individual mask, namely, character-adaptive masking. Next, the recognizer determines the probability of each candidate and decides which candidates are acceptable, and each accepted candidate will start a new processing branch for the latter segmentation; consequently, there will be multiple segmentation solutions. Finally, the solution with the highest holistic recognition probability will be accepted. Compared with existing merged-character recognition approaches, the advantages of the proposed method are the following:

- 1) forepart prediction is verified by reliable character matching;
- 2) NSM algorithm utilizes a concise feature-row-based character model to distinguish similar characters efficiently;
- 3) character masks for the segmentation are adaptive to character shapes so as to avoid damaging the character image.

The rest of this paper is organized as follows. Section II classifies the merging types and reviews the related work. Section III describes the proposed algorithms in detail. The experimental results and comparisons with other methods are reported in Section IV. Finally, Section V draws our conclusions.

II. RELATED WORK

The recognition methods of merged characters can first be divided into two classes: one with segmentation, and one without. The methods without segmentation recognize characters from a text image directly. Most segmentation-free methods are lexicon-based. They treat the word as a single, indivisible entity, and attempt to recognize it using features of the word as a whole. Thus, they are usually called holistic methods. Madhavanath and Govindaraju [4] conduct a survey on holistic methods and summarize “their treatment of lexicon words as distinct pattern classes has traditionally limited their application to recognition scenarios involving small, static

Manuscript received October 19, 2003; revised May 7, 2004. This work was supported in part by the Yau Lee Construction Co. Ltd., Hong Kong, and the Research Grants Council of the Hong Kong Special Administrative Region, China under Grant CUHK 4182/03E. This paper was recommended by Guest Editor V. Govindaraju.

J. Song and M. R. Lyu are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: songj@ieee.org; lyu@cse.cuhk.edu.hk).

Z. Li and S. Cai are with the State Key Laboratory of Novel Software Technology, Nanjing University, Nanjing 210093, China (e-mail: lizuo@graphics.nju.edu.cn; sjcai@netra.nju.edu.cn).

Digital Object Identifier 10.1109/TSMCB.2004.837588

lexicons." Rocha and Pavlidis [5] propose an interesting segmentation-free approach, which converts a text image into a feature graph and then attempts to match subgraphs of features with predefined character prototypes. Different alternatives are represented by a net whose nodes correspond to the matched subgraphs. A final search for the optimal path in the net gives the best interpretation of the text image. Martin *et al.* [6] engage in an input window to scan horizontally over the text image, and train a neural network to recognize whether the window is centered over a single character (if so, the character is recognized) or between characters. Lee and Kim [7] improve this idea by employing the cascade neural network to retain the spatial relationship of connected characters.

Generally, most character recognition methods [8]–[23] include a segmentation process, i.e., first segmenting a given text image into character images and then recognizing each character image separately. OCR methods for the single character recognition have been well studied in the literature [24]. The current directions to improve the OCR performance include designing better classifiers [25], [26], integrating several existing classifiers to utilize various kinds of features [27], and extracting more robust features [28]–[30]. This paper proposes a new feature extraction and matching approach based on FP and NSM. Jung *et al.* [19] also explores the leftmost and rightmost features to recognize merged characters; however, they use linear segmentation, while the proposed method conducts nonlinear segmentation.

Prior to discussing the segmentation approaches, we define the three types of merging relationships: "linear," "nonlinear," and "overlapped" (Fig. 1). In Fig. 1(a), the two merged characters can be completely separated by a linear segmentation path, defined as the "linear" type. Here the "completely separated" means the entirety of both characters is maintained. In Fig. 1(b), the two merged characters cannot be completely separated by any linear path, but can be done by a nonlinear path, defined as the "nonlinear" type. In Fig. 1(c), the two merged characters overlap each other; therefore, they cannot be completely separated by either a linear path or a nonlinear path, defined as the "overlapped" type. Following the definition of merging types, we classify a character segmentation approach into either a linear segmentation approach or a nonlinear segmentation approach, according to the shape of its segmentation path.

Linear segmentation approaches separate merged characters by linear cuts, mostly vertical cuts. They depend on a variety of clues to locate the cutting positions. The most simple and fast way is to analyze the vertical projection of a black region. The representative methods are [8], [9], which locate the valleys of the projection profile as the cutting positions. However, they cannot handle the *linear* type with long vertical connected length [Fig. 2(a)], degraded characters, or italic characters. Tsujimoto and Asada [10] proposed a filter to produce a deeper valley at merging positions using an "AND" operation of adjacent columns, but the improvement is limited. Another popular way is recognition-based segmentation, which generates multiple segmentation hypotheses and then chooses the best one by recognition. The approach of Casey and Nagy [14] recognizes the subimage in a shrinking window whose width decreases leftward [Fig. 2(b)]. Once a character is matched, its

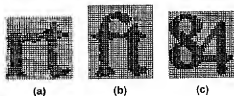


Fig. 1. Three merging types. (a) Linear. (b) Nonlinear. (c) Overlapped.

corresponding block will be cut and the remainders will be processed recursively. This strategy is better than projection-based methods because it selects cutting positions dynamically, but it also cannot handle the *nonlinear* type due to the vertical cutting. Other recognition-based segmentation approaches [15], [19] propose different criteria to select cutting positions, however, they still perform vertical cutting. To decrease the segmentation errors caused by vertical cuts, Bayer *et al.* [11], [12] place a cut classifier before cutting, and perform additional connectivity analysis after cutting to correct the errors. Some connected-component based approaches [13], [18], [22] employ a splitting-and-merging scheme. They first separate the text image into primitive blocks and then combine consecutive blocks into candidate characters using certain criteria, such as knowledge-based dynamic programming [18] and lexicon matching [22]. Recently, Garain and Chaudhuri [23] propose a new approach applying fuzzy multifactorial analysis to identify touching parts and segment merged characters by vertical cuts.

Nonlinear segmentation approaches [16], [17], [20], [21] attempt to separate merged characters with various merging types by the optimal nonlinear cutting paths. Wang and Jean [16] propose an approach called "shortest path segmentation." By predefining the "distance (cost)" needed for selecting the path of each hypothetical cut and then finding the "shortest" one, this method selects the optimal segmentation from all "legal" ones. This approach partially overcomes the problems raised by vertical cutting, and in a way, can adjust the path from vertical cutting. The shortest path consists of several vertical monochromatic runs, as shown in Fig. 3. However, since the adjustment depends only on the local connectivity analysis, it cannot guarantee that the shortest path preserves the character's entirety well. Lee *et al.* [17] extend the shortest path search to the grayscale image space. Arica and Yarmen-Vural [21] define a novel cost function using the information extracted from both grayscale image and binary image, and employ a dynamic programming algorithm to search the shortest path. The above three approaches share the same concept, i.e., searching the shortest path based on the pixel-wise cost accumulation. Chang and Chen [31] introduce the convex-hull information to improve the accuracy of shortest path location. Chen and Wang [20] further propose a different approach which first performs thinning on both foreground and background regions for the feature points extraction on the foreground and background skeletons, then constructs several segmentation paths from these feature points, and finally decides the best segmentation path or rejects the segmentation by the mixture Gaussian probability function.

In summary, existing nonlinear segmentation approaches all depend on local image features to determine segmentation paths, giving no guarantee to maintaining characters' entirety.

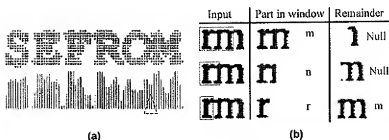


Fig. 2. Various segmentation methods. (a) Projection-based segmentation. (b) Recognition-based segmentation.



Fig. 3. Shortest path segmentation.

The proposed method differs from existing approaches in that the nonlinear segmentation path is formed by character-adaptive masking. We describe our approach in the next section in detail.

III. ALGORITHMS

Correct recognition of merged characters doubtlessly relies on the accuracy of segmentation, which in turn needs image analysis and recognition to avoid the blindness of segmentation. We believe that the segmentation integrating both image analysis and recognition is the most promising way. Considering a horizontal string of merged characters, its leftmost part, i.e., the forepart of the first character, keeps the original shape. Thus, if we can 1) predict a group of candidates by analyzing the forepart, 2) verify the candidates by character recognition, and finally 3) cut out the subimage of the recognized character without damaging the succeeding character, then the entire string of merged characters can be recognized recursively. This is the fundamental idea of our proposed method.

The three conditions reveal the three key techniques of our method: FP-based prediction, NSM-based character recognition, and character-adaptive masking. This section will explain our method in a bottom-up order. Since the character recognition is an elementary process, it will be described first. The prediction and masking algorithms are presented next. Finally, we illustrate the overall paradigm of the merged characters recognition.

A. Character Recognition

Being an elementary process, the character recognition will be invoked very frequently; therefore, it should be time-efficient and accurate. The traditional character bitmap model cannot satisfy this requirement. Letting the prototype binary bitmap of a character be M and the input binary bitmap of a character be X , the resulting binary bitmap of matching M and X is calculated by $R = M \oplus X$. The nonzero points in R are called *different points (DP)*. A *DP* caused by degradation or displacement of the same stroke is called *Edge_DP*, as shown in Fig. 4(a). A *DP* caused by different strokes is called

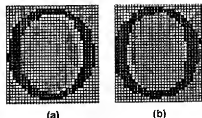


Fig. 4. Examples of DPs. (a) *Edge_DP* $M = "O"; X = "O";$ (b) *Stroke_DP* $M = "O"; X = "U";$

Stroke_DP, as shown in Fig. 4(b). Since the character recognition is normally to detect *Stroke_DPs* and tolerate *Edge_DPs*, judging only by the quantity of *DP* is not robust for degradation and displacement; for example, Fig. 4(a) and Fig. 4(b) contain a similar number of *DPs*. On the other hand, distinguishing *Edge_DP* from *Stroke_DP* requires more complex global analysis. Therefore, the entire bitmap model is not a good choice. Moreover, the entire bitmap model contains much redundancy whose contribution to the recognition is less than the noise it introduces. Thus, we design a new compact feature extraction and matching algorithm.

1) *Feature matching algorithm*: To reduce the processing time and to achieve the maximum marginal utility of character features, the number of features should be the minimal that can distinguish English characters and punctuations. Since the bitmap data is organized row by row, computation in the row direction is fast. Therefore, we define the character feature as a group of rows in the prototype bitmap of a character, called *feature rows*, denoted by f .

Theoretically, the feature matching function should check whether each feature row and its corresponding row in an input bitmap are identical, as described in Definitions 1 and 2.

Definition 1: Let $\text{MATCH}(f, X)$ be the matching function of f and an input bitmap X . If they match, the function returns 1; otherwise, it returns 0.

Definition 2: Let M_i be a row in the prototype bitmap M of a character, where i is the row index. Let M' be the prototype bitmap of another character. A group of rows of M , denoted by $f_m = \{M_i | i = r_1, r_2, \dots, r_k\}$, can be the feature rows of this character on the condition that

$$\text{MATCH}(f_m, M') = \begin{cases} 1, & \text{if and only if } M = M' \\ 0, & M \neq M'. \end{cases} \quad (1)$$

However, since the scanned image often contains degradation or displacement, the condition in (1) is too strict. Consequently,

we propose the necessity-sufficiency matching algorithm based on *dilated feature rows* and *contracted feature rows*.

Definition 3: Let $f_i(t = 1 \dots k)$ be a feature row consisting of several consecutive black and white runs. Then, the dilated feature row of f_i , denoted by f_i^d , is formed by adding two more black points to both the head and the tail of each run. Similarly, the contracted feature row of f_i , denoted by f_i^c , is formed by removing two head points and two tail points from each run; however, each run has to retain at least one point. Thus, $f = \{f_i, f_i^c, f_i^d\} | t = 1 \dots k$.

With Definition 3, if X resembles M , X should cover $f_i^c(t = 1 \dots k)$ completely and be covered by $f_i^d(t = 1 \dots k)$ completely. Suppose that for each character, its prototype bitmap and the input bitmap have been normalized to the same width (W) and height (H). The necessity matching condition is defined as (2), where $row(t)$ returns the row index of t in the bitmap

$$N(f, X) = \sum_t N_t(f_t, X) = 0$$

$$\text{where } N_t(f_t, X) = \sum_{j=1}^W \left[\left(f_{tj}^c \wedge X_{row(t)j} \right) \oplus f_{tj}^d \right]. \quad (2)$$

And, the sufficiency matching condition is defined as (3)

$$S(f, X) = \sum_t S_t(f_t, X) = 0$$

$$\text{where } S_t(f_t, X) = \sum_{j=1}^W \left[\left(X_{row(t)j} \wedge f_{tj}^d \right) \oplus X_{row(t)j} \right]. \quad (3)$$

Thus, we can define the necessity-sufficiency matching function as follows:

$$\text{MATCH}(f, X) = \begin{cases} 1, & \text{if } N(f, X) = 0 \text{ and } S(f, X) = 0 \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

Equation (4) judges whether X matches the feature rows of a character. The binary output is adequate for selecting feature rows based on Definition 2, but it cannot indicate to which extent X resembles the character. Considering this information is important for the segmentation to make decision when multiple solutions exist, we modify the above equations to yield the probability of X being the specific character. Equations (2)–(4) are modified to be (5)–(7), respectively.

$$N'(f, X) = \prod_t [1 - N'_t(f_t, X)], \quad \text{where}$$

$$N'_t(f_t, X) = \frac{1}{W} \cdot \sum_{j=1}^W \left[\left(f_{tj}^c \wedge X_{row(t)j} \right) \oplus f_{tj}^d \right] \quad (5)$$

$$S'(f, X) = \prod_t [1 - S'_t(f_t, X)], \quad \text{where}$$

$$S'_t(f_t, X) = \frac{1}{W} \cdot \sum_{j=1}^W \left[\left(X_{row(t)j} \wedge f_{tj}^d \right) \oplus X_{row(t)j} \right] \quad (6)$$

$$\text{MATCH}'(f, X) = N'(f, X) \times S'(f, X),$$

$$0 \leq N'(f, X) \leq 1 \text{ and}$$

$$0 \leq S'(f, X) \leq 1. \quad (7)$$

Thus, (2)–(4) form the NSM algorithm for selecting feature rows, whereas (5)–(7) form the NSM algorithm for character recognition. An input bitmap X can be recognized as the character with the maximum matching probability.

The normal, dilated and contracted feature rows can tolerate horizontal displacement, but they are font-dependent. This design is based on the consideration that it will enhance the capability of distinguishing similar characters.

2) Selection of feature rows: The process of selecting feature rows for a character consists of two steps 1) preparing the prototype bitmap for this character, and 2) automatically generating candidate feature.

The feature rows are independent from font size since both the prototype bitmap and the input bitmap are normalized. Since they are dependent on font style, we generate separate feature libraries for ten commonly-used fonts. This fashion is suitable for our application—tenders, which require specific fonts in specific parts, so we can preload the feature library of the specific font by tender layout analysis. In each feature library, the prototype bitmap of every character has three variants for regular, bold, and italic styles, respectively.

With the prototype bitmaps of all characters ready, the feature rows of each character are automatically selected by the following four steps. For the current character, denote the prototype bitmap by M , the target feature row set by f_r , and the candidate row set by C_r .

- 1) Initially, $f_r = \{\}$, and C_r contains all rows in M ;
- 2) For each row r in C_r , first make $f'_r(r) = f_r + \{r\}$, then count the times of f'_r matching other prototype bitmaps than M , denoted by M' , in the same feature library: $M_SUM(r) = \text{EMATCH}(f'_r(r), M')$, finally select the r_{\min} with the minimum $M_SUM(r)$. If more than one r reach a tie for the r_{\min} , select the one with the maximum vertical distance to the existing feature rows in f_r to distribute the feature rows evenly;
- 3) Move r_{\min} from C_r to f_r ;
- 4) If $M_SUM(r_{\min}) = 0$, the current f_r is the feature row set of M , then terminate. Otherwise, go to (2).

The automatic feature row selection is offline and one-off; therefore, the time efficiency is not critical. The number of feature rows for each character is different. According to our experiment, the number is at most five. Most characters have four or five feature rows.

Fig. 5 shows a tool for monitoring the feature rows of a selected character. The left side shows the prototype bitmap and the feature rows. The current feature row together with its contracted and dilated feature rows is drawn at the bottom. The right side displays the matched characters of each feature row, and the box “matched chars” gives the characters matching all feature rows. Only the ground-truth character should appear in this box.

An important issue before matching feature rows is the alignment between the input bitmap and the feature rows, especially the vertical alignment since the NSM algorithm can tolerate small horizontal offsets. There are two common alignment methods: bounding-box alignment and gravity-center alignment. The former is simple but sensitive to noise, while the latter is antinoise but computation-expensive. We utilize the

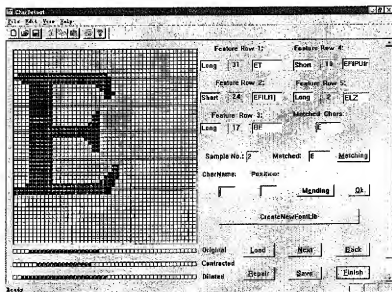


Fig. 5. Feature rows.

baseline of printed text line (Fig. 6) for alignment. The baseline is defined at the bottom of upper case characters. Since the baseline is obtained from a text line, it is more antinoise than the gravity center, and it is fast. Therefore, the baseline position of each character is also labeled in the feature library.

B. FP-based segmentation

The FP-based segmentation algorithm consists of two steps: forepart prediction and character-adaptive masking. The former selects a group of candidates by forepart analysis, while the latter screens out the bitmap using the nonrectangular mask adaptive to each candidate and then recognizes the segmented bitmap by the NSM algorithm. Therefore, the merged characters can be segmented with the highest recognition probability.

1) *Forepart prediction:* After horizontally scanning a document image, we obtain the baseline position and the height of each text line, which is the vertical distance between the baseline and the top of the text line. Denoting the height of text line by H_i , the term "forepart" means the leftmost $H_i/4$ wide part of the input image of merged characters. For a prototype bitmap, it is nearly the left half of the bitmap. The forepart prediction is based on three reliable forepart features, i.e., baseline-related feature, forepart height feature, and forepart boundary feature.

The baseline-related feature indicates whether the forepart of a character has components under the baseline, which takes value "true" or "false." The forepart height feature indicates whether the forepart occupies the full height above the baseline, which also takes value "true" or "false." The forepart boundary feature is a little more complex, defined as follows:

$$B = \{B(i)|i \text{ is the row index of a bitmap} \\ \text{and baseline} \leq i \leq \text{baseline} + H_i\}. \quad (8)$$

$B(i)$ is the number of white points before the first black point in the row i . Considering the input scanned bitmap may contain noises, it can be revised to be the number of points before the first black segment whose width is not

Billing system Baseline

Fig. 6. Baseline of a text line.

less than the minimum stroke width. The difference between two forepart boundary features B_1 and B_2 is defined as $|B_1 - B_2| = \text{MAX}_{\text{baseline} \leq i \leq \text{baseline} + H_i} ||B_1(i) - B_2(i)||$.

These three features of each character are obtained from its prototype bitmap and then stored in the feature library. Denote the baseline-related feature by L , the forepart height feature by G , and the forepart boundary feature by B . Correspondingly, these three features obtained from the forepart of an input bitmap are denoted by L' , G' and B' , respectively. Letting the initial candidate set C_0 containing all models in a feature library, we can select candidates by the following operations:

$$\begin{aligned} C_1 &= \{c \in C_0 | L = L'\} \\ C_2 &= \{c \in C_1 | G = G'\} \\ C_3 &= \{c \in C_2 | |B - B'| < \varepsilon\}. \end{aligned}$$

The dimension of C_3 is much smaller than that of C_0 , however, candidates like "P" and "K" cannot be distinguished. To further reduce the number of candidates, for each candidate in C_3 , we perform a quick necessity matching by (2) to form C_4 , which is the final candidate set produced by the forepart prediction.

$$C_4 = \{c \in C_3 | N(f_c, X) = 0\}.$$

2) *Character-adaptive masking:* The forepart prediction produces a small number of candidates. The best way to verify each candidate is to screen out the bitmap of the leftmost character and recognize it. To avoid the disadvantages of vertical cutting, we propose a character-adaptive masking algorithm, which uses an individual mask for each character. The mask is generated from the prototype bitmap of the character. Before defining the mask, we first define the right boundary of the

character. Recall the width and height of the prototype bitmap are W and H , respectively. The right boundary is defined as follows:

$$RB = \{RB(i) \mid i \text{ is the row index of a bitmap, and } 0 \leq i \leq H\}$$

where $RB(i)$ is the horizontal coordinate of the rightmost black point on the row i . If the entire row is white, $RB(i) = 0$. Then, the real width of the character $W_c = \max_{0 \leq i \leq H} [RB(i)]$. Now we can define the character mask.

Definition 4: The mask is a binary matrix whose row index is the same as that of the prototype bitmap and whose width is W_c . The value of each point in the mask is defined by (9), also as shown in Fig. 7.

$$\text{Mask}(i, j) = \begin{cases} 1, & 0 < j \leq \max(RB(i), \frac{W_c}{2}) \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Notice that the right bound " $\max(RB(i), W_c/2)$ " is important for the sufficiency matching. For example, when the leftmost character is "B," the candidates may include "B," "P," an "L." If simply using $RB(i)$ as the right bound, the masked bitmap of "P" or "L" will also be recognized perfectly. Using the mask of a candidate (denoted by c) to "AND" with the input bitmap pixel by pixel, we can segment the bitmap of the candidate (X_c) from the remainder along the path determined by the right shape of the candidate. Character-adaptive masking is very effective in segmenting *linear* and *nonlinear* merging types. For the *overlapped* type, at least the shape of the left character can be preserved.

C. Merged Character Recognition Paradigm

Compute the recognition probability of c by (7), i.e., $P_c = \text{MATCH}(f_c, X_c)$. If only accepting the candidate with the highest P_c , the segmentation will have one solution; however, it also has a higher risk of rejection or misrecognition of the entire string of merged characters. On the other hand, if accepting every candidate whose P_c is high enough, the segmentation may yield multiple solutions, and the best solution can be selected by the holistic recognition probability. For the best performance, we choose the latter way.

The recognition of a scanned document image first separates the image into several areas by layout analysis, then divides each area into subimages of text lines by horizontal projection, and finally cuts each text line into character bitmaps at zero-height positions of vertical projection profile. The final bitmap may contain either single character or merged characters, which are distinguished by checking the aspect ratio and the side profile [19]. The bitmap of single character is recognized by the NSM algorithm directly, while the bitmap of merged characters becomes the input bitmap X of our recognition paradigm shown in Fig. 8.

Since this paradigm allows multiple solutions, we need to use a stack to store the branchpoint status for the recursive process. The structure of stack element consists of three variables, as shown in the double-bordered box at the top-left corner of Fig. 8. P_h is the holistic recognition probability up to the branchpoint.

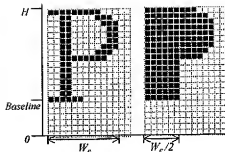


Fig. 7. Mask of "P"

"String" contains a list of recognized characters. "Bitmap" is the current remainder of bitmap to be recognized. Initially, the feature library is ready, and the stack contains only one element, whose P_h is 1, "String" is empty, and "Bitmap" is the input bitmap of merged characters.

The recursive process begins with popping up the top element of the stack. Then, the forepart prediction takes place to select a small number of candidates into C_4 by four procedures. Next, for each candidate c in C_4 , (1) the character-adaptive masking screens out the bitmap of this candidate, denoted by X_c , correctly. (2) The NSM-based character recognition is performed to obtain the recognition probability P_c . (3) Check whether $P_c > \xi$, where ξ is a predefined threshold for accepting the recognition probability. If $P_c \leq \xi$, this candidate is discarded and the following steps are skipped. (4) Check whether the remainder of the input bitmap, i.e., "bitmap", X_c , is empty. If so, all merged characters are processed. Therefore, a solution, consisting of two variables: P_h and "String", is added to the solution array and the following step is skipped. (5) Push the current recognition status into the stack. The value of the stack element representing the current recognition status is assigned as follows: $P_h = P_h \times P_c$, "string" is appended with the current candidate c , and "bitmap" is the remainder after masking. Once all candidates in C_4 have been processed, check whether the stack is empty. If so, the recognition finishes. Otherwise, perform the recursive process again.

After the recursive process finishes, we get n solutions. If $n = 0$, the rejection happens. If $n = 1$, the only solution becomes the final solution. Otherwise, there exist multiple solutions. Usually, the one with the highest holistic recognition probability (P_h) will become the final solution. When the vocabulary of the application is limited, spelling check is also useful.

IV. PERFORMANCE ANALYSIS AND EXPERIMENTS

A. Performance Analysis

Table I presents a performance comparison on segmentation between the proposed method and other representative methods in terms of general characteristics, segmentation accuracy, adaptability to three merging types, and speed. The evaluation is based on the experimental results of implementing the projection-based segmentation approach [8], recognition-based segmentation approach [14], and shortest-path segmentation approach [16].

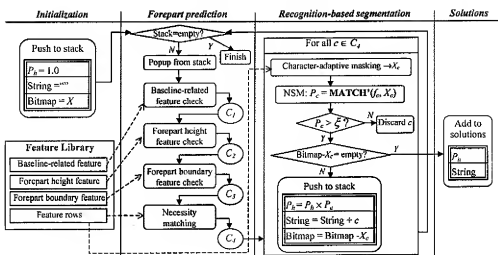


Fig. 8. Merged character recognition paradigm.

TABLE I
PERFORMANCE COMPARISON BETWEEN FOUR SEGMENTATION METHODS

Method	Projection-based Segmentation [8]	Recognition-Based Segmentation [14]	Shortest Path Segmentation [16]	Our method
General Characteristics	Static, vertical cutting path.	Dynamic, vertical cutting path.	Dynamic, nonlinear cutting path.	Dynamic, character-adaptive cutting path.
Accuracy of segmentation	Low	Medium	Relatively high	High
Adaptability	Poor	Low	Medium	High
Linear	Medium	Good	Good	Good
Nonlinear	No	No	Medium	Good
Overlapped	No	No	Limited	Medium
Speed	Fast. Projection analysis requires little computation.	Slow. Trying all possible cutting positions.	Fast, the algorithm of searching path is not complicated.	Medium, most of the time was spent on the forepart prediction.

Table I shows that the segmentation quality of our proposed method is the best among the four methods. The advantages of our method come from the forepart prediction and the character-adaptive masking. For the merged characters of the *linear* and *nonlinear* types, our method can segment them correctly. For those of the *overlapped* type, our method can correctly segment at least the left character, while the other methods cannot guarantee segmenting any of them correctly.

B. Experiments

We implemented the proposed method in an experimental system—VHTender, which processes scanned tenders for construction companies. To evaluate the performance of the proposed algorithms in detail, we divide the experiment into two parts: NSM-based character recognition and FP-based merged character segmentation.

1) *Result of character recognition*: The proposed NSM-based character recognition algorithm is compared with a commercial OCR system—OmniPage Pro 10 (OPP10). We choose OPP10 for two reasons: 1) OPP is a well-known OCR software with a long development history. Its developer,

ScanSoft company, claims that OPP10 is one of the best OCR system in the world, 2) The testing documents in our experiment are construction tenders, which contain a lot of underlined texts. Many OCR systems cannot handle underlined texts well, especially when text touches its underline. However, OPP10 is not very sensitive to it.

The testing data are the scanned images of 12 tender pages, containing 12 786 characters. Table II shows the recognition results of two systems obtained in a PC (PIII500/256M).

The number of errors includes both false recognition errors and rejection errors. We analyze the cause of each error and report the error distributions of two systems in Table III.

The experimental results show that both systems achieve high recognition rates. They have similar performances on degraded characters and merged characters; however, VHTender has a much better ability in distinguishing characters that are similar in shape, e.g., “1-1-1,” which makes VHTender achieve a higher recognition rate. This advantage of VHTender comes from that the feature rows of the similar characters are automatically defined at the location of the maximum difference and the NSM algorithm is sensitive to the difference of even one feature row. The overall processing time (T_{all}) of VHTender is longer than

TABLE II
PERFORMANCE COMPARISON BETWEEN VHTENDER AND OPP10

Item System	Number of errors	Recognition rate	T _{all} (sec/page)	T _{OCR} (sec/page)	Matching function speed (sec)
OPP10	88	99.31%	0.94	N.A	N.A
VHTender	37	99.71%	1.21	0.24	0.89×10^{-4}

TABLE III
RECOGNITION ERROR DISTRIBUTIONS OF TWO SYSTEMS

Reason System	Similar in shape	Wrong segmentation	Degradation	Unknown	Total
OPP10	57	12	7	12	88
VHTender	16	10	11	0	37

that of OPP10. Since VHTender is a system designed for automatic tender analysis, T_{all} includes the time for layout analysis and tender understanding. The pure time for OCR (T_{OCR}) is only one fifth of T_{all} .

2) *Result of character segmentation*: We select another 20 pages containing heavily merged characters to test the FP-based segmentation algorithm. Fig. 9 shows the nature of the testing image. The total number of merged characters and the distribution of merging types are given in Table IV.

Our FP-based segmentation algorithm (abbreviated to FPS) is compared with the shortest path segmentation algorithm [16] (abbreviated to SPS). For easy understanding, we describe the SPS algorithm briefly. A candidate cutting path P_i starts from each column in the bottom (or top) row of an input bitmap, then selects the least cost for P_i to move to the next row upward (or downward), and finally stops when reaching the top (or bottom) row. At any row, P_i can only move to one of its left-diagonal, vertical, or right-diagonal neighbors in the next row. The cost for a move is defined as follows: a vertical move to a white pixel costs 0, a vertical move to a black pixel costs 10, a diagonal move to a white pixel costs 1, and a diagonal move to a black pixel costs $10\sqrt{2}$. The SPS algorithm uses a one-dimensional (1D) cost array $\{C_i\}$ to record the accumulative cost for each P_i and a two-dimensional (2-D) path array to store the route of each P_i . After obtaining the costs of all candidate paths, the path with the least cost, i.e., the shortest path, will be considered as the cutting path.

Some examples of the segmentation results of the two algorithms are shown in Fig. 10, where the cutting paths are drawn in gray. The cutting path of the SPS algorithm consists of 8-connected vertical lines, whereas that of the FPS algorithm follows the right boundary of the character mask. The left four examples are all with the *Linear* merging type and the vertical length of connected parts are very short. Both two algorithms segment them well with small difference, such as the bottom-left corner of "a" in the example "na," which demonstrates the FPS algorithm can preserve the shape of character better than the SPS algorithm. For the *nonlinear* merging type, e.g., "ob" and "on" in the right examples, the FPS algorithm does a better clean cut than the SPS algorithm. The abilities of two algorithms differs much in handling the *overlapped* merging types. In the example "omp," "om" is with the *nonlinear* type and "mp" is with the *overlapped* type. Since the lengths of con-

complete the panel wall part alternatives in accordance w to obviate movement and any joints/junctions where surfa

Fig. 9. Fraction of testing image.

nected parts are long, the cost of cutting through the black areas becomes very high. Consequently, the SPS algorithm mis-segments "omp" into "cnp." On the contrary, the FPS algorithm segments three characters correctly by using forepart prediction and character-adaptive masking. The only imperfection of the FPS result is that "p" loses those pixels overlapping with "m."

The analysis on the least cost profile of SPS (Fig. 11) reveals the three following theoretical disadvantages of the SPS algorithm:

- 1) definition of cost prevents the cutting path from passing many black pixels, which, however, is a must to segment long connected parts;
- 2) usually, the shortest path is not the best cutting path. Sometimes, many paths have similar costs. Therefore, the SPS algorithm has to invoke the character recognition to select the best path, which somewhat diminishes the advantage of "shortest path.;"
- 3) least cost profiles generated by bottom-up accumulation and top-down accumulation are not consistent.

For example, the top-down cost profile provides a better path to segment "ve" rather than the bottom-up profile. However, the SPS algorithm did not consider this inconsistency. Compared to the "shortest path" concept, the segmentation based on forepart prediction is more straightforward and goal-directed.

Table IV reports the quantitative experimental results, which demonstrates that the FP-based segmentation achieves remarkable improvement when handling the *nonlinear* and *overlapped* merging types. The improvement attributes to three factors: the effective forepart prediction, the character-adaptive masking and the accurate NSM-based character recognition.

V. CONCLUSION

This paper classifies the merging relationship between two involved characters into three types: *linear*, *nonlinear*, and *overlapped*. Existing segmentation methods usually handle the *linear* type well; however, their capabilities of handling

TABLE IV
EXPERIMENTAL RESULT OF FP-BASED SEGMENTATION

Type	Linear	Nonlinear	Overlapped	Total
Number of characters	805	834	14	1653
FP-based segmentation	Error_Num	10	12	28
	Accuracy	98.8%	98.6%	98.3%
Shortest path segmentation	Error_Num	13	10	184
	Accuracy	98.4%	80.7%	28.6%

Original bitmap	Segmented by SPS	Segmented by FPS	Original bitmap	Segmented by SPS	Segmented by FPS
na	na	na	obv	obv	obv
su	su	su	omp	omp	omp
th	th	th	ons	ons	ons
wa	wa	wa	pan	pan	pan

Fig. 10. Segmentation comparison between SPS and FPS.

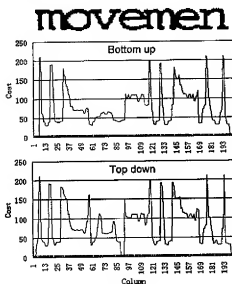


Fig. 11. Least cost profile of SPS.

the other two types are rather limited. This paper proposes a novel merged-character recognition method based on forepart prediction, NSM-based recognition, and character-adaptive masking. This method handles both the *linear* and the *non-linear* merging types well. For the *overlapped* type, the left character is segmented correctly at the cost of possibly breaking the entirety of the right character. In summary, this paper makes two main contributions. The first one is the NSM-based character recognition. This algorithm utilizes feature rows to reduce the redundancy of character feature such that the feature rows are selected to maximize the difference among similar characters; therefore, this algorithm is fast and accurate. Due to the use of dilated and contracted feature rows, it is also antinoise and able to tolerate one- or two-pixel horizontal displacement between the character model and the character bitmap to be matched. The second contribution is the FP-based

segmentation using forepart prediction and character-adaptive masking. The effective forepart prediction quickly eliminates most irrelevant candidates, and the character-adaptive masking, making this segmentation algorithm the first one engaging character-dependent cutting path, is very important to preserve the shapes of merged characters. The experimental results and the performance comparisons with other methods demonstrate that the proposed method achieves remarkable improvement over existing methods.

REFERENCES

- [1] O. D. Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition—a survey," *Pattern Recognit.*, vol. 29, no. 4, pp. 641–662, 1996.
- [2] T. Nariker, "ISRI Annual Report," Univ. Nevada, Las Vegas, NV, 1993.
- [3] R. G. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 690–706, Jul. 1996.
- [4] S. Madhavan and V. Govindaraju, "The role of holistic paradigms in handwritten word recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 149–164, Feb. 2001.
- [5] J. Rocha and T. Pavlidis, "Character recognition without segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 9, pp. 903–909, Sep. 1995.
- [6] G. L. Martin, M. Rashid, and J. A. Pittman, "Integrated segmentation and recognition through exhaustive scans or learned saccadic jumps," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 7, no. 4, pp. 831–847, 1993.
- [7] S.-W. Lee and S.-Y. Kim, "Integrated segmentation and recognition of handwritten numerals with cascade neural network," *IEEE Trans. Syst. Man, Cybern. C, Appl. Rev.*, vol. 29, no. 2, pp. 285–290, May 1999.
- [8] Y. Lu, "On the segmentation of touching characters," in *Proc. Int. Conf. Document Analysis Recognition (ICDAR)*, Tsukuba City, Japan, Oct. 1993, pp. 440–443.
- [9] S. Liang, M. Ahnadi, and M. Shridhar, "Segmentation of touching characters in printed document recognition," in *Proc. Int. Conf. Document Analysis Recognition (ICDAR)*, Tsukuba City, Japan, Oct. 1993, pp. 569–572.
- [10] S. Tsujimoto and H. Asada, "Major components of a complete text reading system," *Proc. IEEE*, vol. 80, no. 7, pp. 1133–1149, July 1992.
- [11] T. Bayer, U. Krebel, and M. Hammelsbeck, "Segmenting merged characters," in *Proc. 11th Int. Conf. Pattern Recognition (ICPR)*, 1992, pp. 346–349.
- [12] T. Bayer and U. Krebel, "Cut classification for segmentation," in *Proc. Int. Conf. Document Analysis Recognition (ICDAR)*, 1993, pp. 565–568.

- [13] G. Seni and E. Cohen, "External word segmentation of off-line handwritten text lines," *Pattern Recognit.*, vol. 27, no. 1, pp. 41-52, 1994.
- [14] R. G. Casey and G. Nagy, "Recursive segmentation and classification of composite patterns," in *Proc. 6th Int. Conf. Pattern Recognition (ICPR)*, 1982, pp. 1023-1026.
- [15] H. Fujisawa, Y. Nakano, and K. Kurino, "Segmentation methods for character recognition: from segmentation to document structure analysis," *Proc. IEEE*, vol. 80, no. 7, pp. 1079-1092, Jul. 1992.
- [16] J. Wang and J. Jean, "Segmentation of merged characters by neural networks and shortest path," *Pattern Recognit.*, vol. 27, no. 5, pp. 649-658, 1994.
- [17] S.-W. Lee, D.-J. Lee, and H.-S. Park, "A new methodology for gray-scale character segmentation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 1045-1050, Oct. 1996.
- [18] L. Y. Tseng and R.-C. Chen, "A new method for segmenting handwritten Chinese characters," in *Proc. Int. Conf. Document Analysis Recognition (ICDAR)*, 1997, pp. 568-571.
- [19] M.-C. Jung, Y.-C. Shin, and S. N. Srihari, "Machine printed character segmentation method using side profiles," in *Proc. IEEE Int. Conf. Systems, Man, Cybernetics (SMC)*, vol. 6, 1999, pp. 863-867.
- [20] Y.-K. Chen and J.-F. Wang, "Segmentation of single- or multiple-touching handwritten numeral string using background and foreground analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1304-1317, Nov. 2000.
- [21] N. Arica and F. T. Yarmar-Vural, "Optical character recognition for cursive handwriting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 801-813, Jun. 2002.
- [22] C.-L. Liu, M. Koga, and H. Fujisawa, "Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1425-1437, Nov. 2002.
- [23] U. Garain and B. Chaudhuri, "Segmentation of touching characters in printed Devanagari and Bangla scripts using fuzzy multifactorial analysis," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 4, pp. 449-459, Nov. 2002.
- [24] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical review of OCR research and development," *Proc. IEEE*, vol. 80, pp. 1029-1058, Jul. 1992.
- [25] S.-B. Cho, "Recognition of unconstrained handwritten numbers by double self-organizing neural network," in *Proc. 13th Int. Conf. Pattern Recognition*, Vienna, Austria, 1996, pp. 426-430.
- [26] C.-T. Chang and L. Y. Tseng, "A heuristic algorithm for the recognition of printed Chinese characters," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 4, pp. 710-717, Apr. 1995.
- [27] T. K. Ho, J. J. Hull, and S. N. Srihari, "Decision combination in multiple classifier system," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 1, pp. 66-75, Jan. 1994.
- [28] I.-S. Oh, J.-S. Lee, and C. Y. Suen, "Analysis of class separation and combination of class-dependent features for handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 1089-1094, Oct. 1999.
- [29] I. S. Oh and C. Y. Suen, "A feature for character recognition based on directional distance distribution," in *Proc. 4th Int. Conf. Document Analysis Recognition*, 1997, pp. 288-292.
- [30] K. Yamada, "Non-uniformly sampled feature extraction method for Kanji character recognition," in *Proc. 4th Int. Conf. Document Analysis Recognition*, 1997, pp. 200-205.
- [31] T.-C. Chang and S.-Y. Chen, "Character segmentation using convex-hull techniques," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 13, no. 6, pp. 833-858, 1999.



Zuo Li received the B.S. degree in physics from Peking University, Beijing, China, in 1988 and the Ph.D. degree in computer science and application from Nanjing University, Nanjing, China, in 2001. Currently, he is a Member of the Technical Staff of Bell Labs, Lucent Technologies, Beijing, China. His research interests include graphics recognition and mobility intelligent network solution.



Michael R. Lyu (S'84-M'88-SM'97-F'04) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 1981, the M.S. degree in computer engineering from University of California, Santa Barbara, in 1985, and the Ph.D. degree in computer science from University of California, Los Angeles, in 1988.

He is currently a Professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. He was with the Jet Propulsion Laboratory, Pasadena, CA, as a Technical Staff Member from 1988 to 1990. From 1990 to 1992, he was with the Department of Electrical and Computer Engineering, The University of Iowa, Iowa City, as an Assistant Professor. From 1992 to 1995, he was Member of the Technical Staff in the applied research area of Bell Communications Research/Bellcore, Morristown, NJ. From 1995 to 1997, he was Research Member of the Technical Staff at Bell Laboratories, Murray Hill, NJ. His research interests include software reliability engineering, distributed systems, fault-tolerant computing, mobile networks, Web technologies, multimedia information processing, and E-commerce systems. He has published over 190 refereed journal and conference papers in these areas. He has participated in more than 30 industrial projects, and helped to develop many commercial systems and software tools. He was the editor of two book volumes: *Software Fault Tolerance* (New York: Wiley, 1995) and *The Handbook of Software Reliability Engineering* (Piscataway, NJ: IEEE and New York: McGraw-Hill, 1996). He is Associate Editor for the *Journal of Information Science and Engineering*.

Dr. Lyu received the Best Paper Awards ISSRE in 1998 and 2003, respectively. He served on the Editorial Board of IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, and has been an Associate Editor of IEEE TRANSACTIONS ON RELIABILITY. He initiated the First International Symposium on Software Reliability Engineering (ISSRE) in 1990. He was the program chair for ISSRE'96, and has served in program committees for many conferences, including ISSRE, SRDS, HASE, ICCETS, ISIT, FTCS, DSN, ICDSN, EUROMICRO, APSEC, PRDC, PSAM, ICCCN, ISESE, and WWW. He was the General Chair for ISSRE2001, and the WWW10 Program Co-Chair. He has been frequently invited as a keynote or tutorial speaker to conferences and workshops in U.S., Europe, and Asia.



Jiqiang Song (S'99-M'02) received the B.S. degree in computer science and the Ph.D. degree in computer science and application from Nanjing University, Nanjing, China, in 1996 and 2001, respectively. Currently, he is a Postdoctoral Fellow at the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin. His research interests include graphics recognition, automatic interpretation of engineering drawings, image/video processing, and video compression. He has published over 30 papers in these areas.



Shijie Cai received the B.S. degree in mathematics, from Nanjing University, Nanjing, China, in 1967. He is a Professor at the Department of Computer Science and Technology, Nanjing University. His research interests include computer graphics, graphics recognition, image processing, and document analysis and recognition.

Semantic Home Photo Categorization

Seungji Yang, Sang-Kyun Kim, and Yong Man Ro, *Senior Member, IEEE*

Abstract—A semantic categorization method for generic home photo is proposed. The main contribution of this paper is to exploit a two-layered classification model incorporating camera metadata with low-level features for multilabel detection. The two-layered support vector machine (SVM) classifiers operate to detect local and global photo semantics in a feed-forward way. The first layer aims to predict likelihood of predefined local photo semantics based on camera metadata and regional low-level visual features. In the second layer, one or more global photo semantics is detected based on the likelihood. To construct classifiers producing a posterior probability, we use a parametric model to fit the output of SVM classifiers to posterior probability. A concept merging process based on a set of semantic-confidence maps is also presented to cope with selecting more likelihood photo semantics on spatially overlapping local regions. Experiment was performed with 3086 photos that come from MPEG-7 visual core experiment two official databases. Results showed that the proposed method would much better capture multiple semantic meanings of home photos, compared to other similar technologies.

Index Terms—Camera metadata, image classification, photo album, support vector machine.

I. INTRODUCTION

THE GOAL of semantic image categorization is to discover the image semantics from a domain of some given predefined concepts, such as building, waterside, landscape, cityscape, and so forth. Recently, as it is affordable to keep a complete digital record of one's whole life, the need for semantic categorization has been raised in both organizing and managing personal photo collection for minimizing user's manual efforts.

Conventionally, many researches have advanced semantic image indexing and categorization in recent decades [1]–[9]. They mostly focused on reducing the semantic gap between low-level visual features and high-level semantic descriptions, which are closer to human visual perception. Herein, one primary tackling point is learning approach itself so that classifier realizes minimal bound of error in real application. In particular, statistical learning approaches, such as Bayesian probability model [32], Markov random fields (MRF) [1], and support vector machines (SVM) [2], have been successfully employed for semantic categorization. A statistical learning

process commonly includes three steps: 1) observing a phenomenon in the real world; 2) constructing a model of the phenomenon; and 3) making predictions using the model, step by step. A useful approach to build the model of a classifier is to employ discriminative features besides low-level visual features or any combination of both. J. Smith *et al.*, in [7], has proposed semantic image/video indexing using semantic model vectors that are constructed from multiple low-level features where the model vector stands for a set of numerical degrees of strength in relation to different semantic meanings. For better classification, in [2] and [9], spatial image context features have been coupled with low-level features as well.

Human beings sense many levels of visual semantics in photo. However, semantic labels to be discovered are generally limited in a specific domain according to application, due to uncertainty and infinity of semantic knowledge of human beings. Although semantic object segmentation has been implemented by a wide range of approaches for last two decades [33]–[36], how to detect multiple semantic concepts in image is still challenging problem due to low performance and high computational cost.

The problems in semantic categorization can be simplified by using multilayered rather than single-layered approach. Having multiple layers in classification often help to solve a classical image understanding problem that requires effective interaction of high-level semantics and low-level features. The way human beings perceive semantic knowledge of an image is hierarchical. In other words, human beings firstly sense rough, rather simple semantic objects, and then compound them to understand more comprehensively detailed semantic meanings of the image. This sensory mechanism can be imitated by a multilayered learning way. Multilayered approach usually forms a specific hierarchy of layers with one or more classifiers. A classifier in the lower layer aims to capture simple semantic aspects by using low-level features while a classifier in the higher layer interprets more complex semantic aspects by using high-level semantic features. Many researchers have employed the multilayered approaches to semantic categorization [5], [6], [9], [13]–[15].

One state-of-the-art classification method is using SVM [10], [11]. Many conventional classifiers have targeted empirical risk minimization (ERM). But, ERM only utilizes the loss function defined for a classifier and is equivalent to Bayesian decision theory with a particular choice of prior. Thus, an ERM approach often leads to an over-fitted classifier, i.e., classifier is usually too much adapted only to training data. Unlike ERM, structural risk minimization (SRM) minimizes generalization error. The generalization error is bounded by the sum of training set error and a term depending on Vapnik–Chervonenkis (VC) dimension of the learning machine. High generalization can be achieved by minimizing the upper bound. SVM is based on the idea of SRM. The generalization error of SVM is related not to the input dimensionality of the problem, but to the margin with separating

Manuscript received May 6, 2006; revised November 20, 2006. This paper was recommended by Guest Editor E. Izquierdo.

S. Yang and Y. M. Ro are with the Information and Communications University (ICU), 103-6 Daejeon, South Korea (e-mail: yangzeno@icu.ac.kr; yro@icu.ac.kr).

S.-K. Kim is with the Samsung Advanced Institute of Technology (SAIT), 14-1 Gyeonggi, South Korea (e-mail: skkim@sa.it.samsung.co.kr).

Color versions of one or more of the figures in this paper are available online at <http://icccsplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2007.890829

the data. This explains why SVM can have a good performance even in problems with a large number of inputs. So far, SVM has been applied successfully to a wide range of problems.

Unfortunately, naïve SVM is inappropriate for multilayered classifier because the output of the SVM should be a calibrated posterior probability so as to enable post-processing. Basically, SVM is a discriminative classifier, not based on any generative model. Kernel function in the SVM acts as the only distance metric. For the use of SVM ensembles in multilayer, the output of all classifiers in a certain layer should be probabilistically modeled before being used as the input of all classifiers in the next layer. A few studies have tackled this problem [16], [17]. In [16], a parametric model was proposed to fit the SVM output to the posterior probability, instead of estimating class-conditional density. The parameters of the model are adapted to give the best probability output. In [17], the parametric model studied in [16] has been improved for more optimized implementation. It solved the problem that the implementation of the conventional parametric model might not converge to the minimum solution. Although the new method increases complexity, it gives better convergence properties.

Nevertheless, reliably capturing high-level visual semantics with low-level features remains a challenge to real application due to low performance and lack of generality. Unlike image, photo usually includes its camera metadata as well as pixel data itself. Camera metadata can be obtained from Exif header from photo file [18]. In order to take best pictures in a given condition, a camera device is often adapted to surrounding light condition and subjects to be focused. Since camera metadata records this optimal picturing condition for a photo, it is of great benefit to semantic photo categorization in that it provides several useful cues that are independent on pixel data, thus giving some extra knowledge. At the early stage of utilization of camera metadata, taken date/time stamp has been successfully employed to cluster a sequence of unlabeled photos by meaningful event or situation groups [19]–[24]. Especially in [19] and [20], taken date/time stamp feature and color features have been combined together to cluster photos by events in an automatic manner. In general, user demand for event indexing tends to exhibit little coherence in terms of low-level visual features, though camera metadata could help to organize photos in more semantically meaningful event groups. We also developed, in our previous studies [23], [24], a situation-based photo clustering approach by combining taken-date/time stamp of camera metadata and low-level visual features, where situation stands for similar background scenery taken in a close proximity of time.

Especially for semantic photo categorization, Boutell *et al.* in [25], have proposed a probabilistic approach to incorporate camera metadata with low-level visual features. They exploited a useful set of camera metadata, which is related to flash, scene brightness, subject distance and focal length, and verified it in indoor/outdoor, sunset/non-sunset, and man-made/natural scene classification. Boutell's method has one major disadvantage on the applications to generic photo classification. As presented in the study, it has limited application to a few global scenes since it used only global features. A photo usually contains many local semantics that are located in a local region. So, to extend the use of camera metadata to the classification of many other local and

global photo semantics, camera metadata is strongly required to be incorporated with visual features of local photo region. For example, let us see a photo containing a human face in foreground behind background scenery. If its camera focus is on the person, then subject distance and focal length would be probably short. Given this condition, Boutell's classifier may have a difficulty of detecting background scenery in spite of the additional use of low-level visual features for the classification.

Many literatures have dealt with multilayered approaches for multilabel classification. In particular, multilayered SVM has been also adopted for multilabel image classification. However, multilabel detection problem with camera metadata is missing, up to the present. Thus, one remaining challenge for multilabel detection in photo is to model classifiers in the multilayered scheme. Here, the classifiers should be to incorporate regional low-level features with camera metadata so that they can capture multiple local photo semantics. From this idea, in this paper, we exploit a novel multilayered classification model that combines camera metadata, low-level features, and intermediate level of semantic features, especially for multilabel classification. The proposed method is composed of two-layered SVM classifiers. The two-layered SVM classifiers operate to discover local and global photo semantics in a feed-forward way. The first layer aims at measuring the likelihood rate of predefined local photo semantics based on both camera metadata and local low-level visual features. Local photo semantics provide an intermediate level of photo semantics as bridging the semantic gap between low-level features and global, i.e., high-level, photo semantics. In the second layer, we determine one or more potential global photo semantics based on the likelihood ratio of local semantics. To construct a SVM classifier producing a posterior probability, a parametric model is applied to fit the output confidence of the SVM classifier to posterior probability. We also exploit concept merging based on a set of semantic-confidence maps in order to cope with selecting more likelihood local photo semantics on overlapping photo regions.

II. SCHEMATIC MODEL OF HOME PHOTO CATEGORIZATION

A. Overview

Fig. 1 illustrates the schematic model of the proposed home photo categorization, given by two separate approaches. One approach is to incorporate camera metadata, such as aperture number, exposure time, flash-fired, etc., with low-level features, such as color, texture, shape, and so forth. Camera metadata is not obtained from the photo content itself; rather from picturing devices such as cameras and camcorders. The other approach is to stepwise model photo semantics in a way of combining local and global semantics, and to use the models to classify photos into multiple semantic concepts. For this, we predefine a set of local and global semantic concepts. The local semantic concept stands for lower level photo semantics while the global concept stands for higher level of photo semantics. A global concept can be linked with one or more local concepts. The strength of the linkage is represented a probability. In general, the use of lots of probability models in multilayered structure often causes error propagation due to the multiplicative of probabilities. But, the

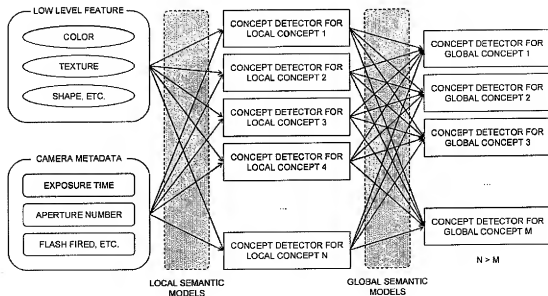


Fig. 1. Proposed two-layered categorization scheme.

use of different feature space for each layer alleviates the error propagation.

Camera metadata is considered at the level of local semantic classification so that it copes with regional low-level features. In general home photo, visual semantics are exhibited on different spatial regions. If camera metadata is combined with low-level features extracted from a holistic photo region, they might be strongly associated with only dominant low-level features of the photo, thus facilitating the detection of a single major semantic. This could be a problem on multilabel categorization. In order to solve the aforementioned problem, we embody the proposed home photo categorization method as a two-layered approach composed of local and global semantic classification processes.

Fig. 2 shows overall procedure of the proposed home photo categorization scheme. The proposed method consists of eight steps. 1) If a photo to be classified is input. 2) then its camera metadata is extracted from the photo file. 3) Entire region of the photo is spatially divided into several local regions. 4) Multiple low-level visual features are extracted from the local regions. 5) Each local region is represented by a combined feature vector constructed with its camera metadata and low-level features. 6) Then, the local features are fed into local semantic classification to detect local photo semantics by measuring joint probability of the local regions features about the models of local semantic concepts. 7) A local semantic feature vector is generated by a concept merging process that makes the local regions keep the highest probable concepts. 8) The local semantic feature vector is fed into global semantic classification to detect global photo semantics by measuring joint probability of the local semantic features about the models of global semantic concepts.

B. Camera Metadata for Home Photo Classification

Camera metadata is usually provided from the Exif header of photo. Most current digital cameras support Exif specification for the camera metadata. It is only available for JPEG photos.

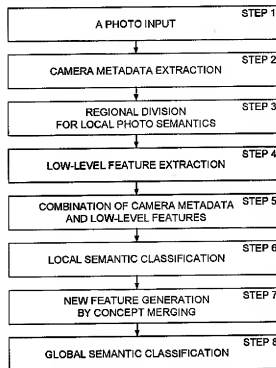


Fig. 2. Overall procedure of the proposed home photo categorization.

But, this is not matter since the JPEG is the most common photo format supported in all consumer cameras. The Exif includes hundreds of metadata tags. Some of the Exif information, such as exposure time, shutter speed, aperture number, flash setting, subject distance, and focal length, are discriminative in specific classes [20]. In sense of usefulness, the camera metadata can be grouped into three families: scene brightness, subject distance, and flash setting (absence or presence). The three classes can be

considered mutually independent each other. Scene brightness includes three tags: exposure time, aperture number, and shutter speed. In general, outdoor sunlighting is brighter than indoor artificial lighting. The exposure-time is closely correlated with the shutter speed. Both exposure-time and shutter speed would become shorter at a brighter scene while longer at a darker scene. The subject distance can be estimated by a camera in auto-focus mode, which becomes usually longer at landscape scenes while shorter at object-oriented scenes. However, the subject distance is dependent of the position of main subjects, so it seems rather useless for general scene classification. For example, the subject distance is short when camera focus is on a certain foreground subject with landscape in the background. Intuitively, the flash setting seems useful in that it would be much more often used in case of lack of lighting such as indoor and night scenes.

Our major concern about the use of camera metadata in photo categorization is a way to combine with other features, such as low-level features. The camera metadata usually presents global scene characteristics although some high-end cameras allow to preserving light condition restricted on some parts of the whole picture. In multilabel detection, many image semantics could be exhibited over small subregions of photo. This means that the main subjects of photographer's interest might be on a small subregion. The low-level features of subregion are definitely useful to derive high-level photo semantics. Accordingly, the camera metadata should be considered to be combined with the regional low-level features to better derive the high-level photo semantics. The proposed two-layered scheme enables to realize this combination as well as to preserve advantages of the use of layered photo semantic features, i.e., local and global semantic features, in multilabel photo classification.

III. LOCAL SEMANTIC CLASSIFICATION

A. Regional Division for Local Semantics

Most current digital cameras support auto-focusing (AF) mechanism that works as moving the camera lens in and out until the sharpest possible image of the subjects is projected onto the image receptor. AF system provides a certain number of censoring regions, of which some regions of interest can be selected by photographer. A censoring region typically forms a rectangle. It means that photographer's intention in capturing photo may be found in the rectangular censoring regions.

The best representation of local image semantics is given by a semantic object segmentation, which produces elaborate object contours or details. In [30], object regions have been found in a hierarchical way based on a partition tree. But, up to the present, there seems no almighty method for object segmentation. Most related approaches are quite expensive in computation and even sometimes produce incomplete result in complex natural images.

So, instead, we propose a simple block segmentation to capture visual semantics that appear on local photo regions. The block segmentation is relatively inexpensive. However, to overcome low segmentation performance, we design a finite set of region templates, in sense of the rectangle censoring system of

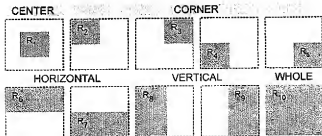


Fig. 3. Photographic region templates.

digital camera. Fig. 3 shows proposed photographic region template, denoted as PRT. Although the PRT is built by block tessellation with a fixed number of blocks, we presume that it could be fast and good enough to capture photographer's intention. The fundamental observation behind the PRT is that mainly concerned subjects are typically focused, taking a larger portion and being sharper than other unconcerned subjects. Thus, many other most likely small, blurred subjects are often out of concern in the photo. In order to build meaningful region templates, three conditions are considered: 1) the PRT should be large enough to detect visual semantics in the local photo region; 2) it should be small enough not to be time-consuming in feature extraction; and 3) it should support spatial scalability to detect semantic subjects in a wide range of spatial scale.

The region template is composed of ten local photo regions: one center region (R_1 in Fig. 3), four corner regions (R_2 , R_3 , R_4 , and R_5 in Fig. 3), two horizontal regions (R_6 and R_7 in Fig. 3), two vertical regions (R_8 and R_9 in Fig. 3), and a whole photo region (R_{10} in Fig. 3). The four corner regions are parts of the vertical, horizontal, and the whole photo regions. And the center region overlaps partially with the corner, vertical and horizontal regions, and entirely with the whole photo region. In this paper, non-overlapping region elements including one center and four corner regions are referred to as "basis regions." Fig. 4 shows some examples with different photographic composition, which are localized by the region templates. In the first example on the left side, the photo could be well-represented using three local regions: the top-vertical region showing sky, left-bottom corner region showing lake, and right-bottom corner region showing field, where the sky, lake, and field can be considered as local semantic concepts of the first example photo.

B. Integration of Camera Metadata and Local Visual Features

To integrate camera metadata with local visual features in the proposed photo categorization, we first generalize the following probabilistic feature combination. Let $\mathbf{X} = \{x_1, x_2, \dots, x_N\}$ be a finite set of N local semantic concepts assumed to frequently appear in home photos. And, let $\mathbf{F}_{\text{cam}} = \{f_{\text{cam}}^1, f_{\text{cam}}^2, \dots, f_{\text{cam}}^J\}$ be a finite set of I camera metadata and $\mathbf{F}_{\text{local}}^{\text{low}} = \{f_{\text{low}}^1, f_{\text{low}}^2, \dots, f_{\text{low}}^J\}$ be a finite set of J low-level features obtained from local photo regions. Then, the likelihood rate of a semantic class of $x_n \in \mathbf{X}$ given feature vector $\mathbf{F}^{\text{local}} = \{\mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{local}}^{\text{low}}\}$ can be denoted as the joint conditional probability like $P(x_n | \mathbf{F}^{\text{local}}) = P(x_n | \mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{local}}^{\text{low}})$.

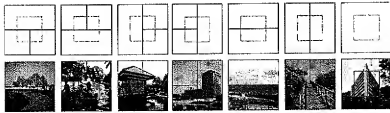


Fig. 4. Examples of local regions segmented by the proposed region template.

By the Bayesian theorem, the joint conditional probability can be decomposed as follows:

$$P(x_n | \mathbf{F}^{\text{local}}) = P(x_n | \mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{low}}^{\text{local}}) \\ = \frac{P(x_n) \cdot P(\mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{low}}^{\text{local}} | x_n)}{P(\mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{low}}^{\text{local}})}. \quad (1)$$

Here, the camera metadata is independent of the regional low-level visual features. So (1) can be written again as follows:

$$P(x_n | \mathbf{F}^{\text{local}}) = \frac{P(x_n) \cdot P(\mathbf{F}_{\text{cam}} | x_n) \cdot P(\mathbf{F}_{\text{low}}^{\text{local}} | x_n)}{P(\mathbf{F}_{\text{cam}}) \cdot P(\mathbf{F}_{\text{low}}^{\text{local}})}. \quad (2)$$

This probability derivation presents the basic methodology for integration of camera metadata and regional low-level visual features. The integrated feature is used only in the first layer, so that we can predict likelihood in relation of a predefined set of local photo semantics. A local semantic feature is presented by degree of strength of the local photo semantics in the first layer.

A comparative probability of $P(x_n | \mathbf{F}^{\text{local}})$ can be $P(x_n | \mathbf{F}_{\text{low}}^{\text{global}})$, which is denoted as $P(x_n | \mathbf{F}_{\text{cam}}, \mathbf{F}_{\text{low}}^{\text{global}})$ where $\mathbf{F}_{\text{low}}^{\text{global}}$ is a finite set of low-level features obtained from a holistic photo region. As studied in the Introduction section, the representation and understanding of local semantics is necessary for multilabel detection. For $\forall x_n, x_n \in \mathbf{X}$, since x_n stands for a local semantic concept implying low-level photo semantics, $\mathbf{F}_{\text{low}}^{\text{local}}$ would have better representation of local photo semantics than would $\mathbf{F}_{\text{low}}^{\text{global}}$. Accordingly, we expect that $P(x_n | \mathbf{F}^{\text{local}}) > P(x_n | \mathbf{F}_{\text{low}}^{\text{global}})$ whenever x_n belongs to the true category that is located on the subregion of the local low-level features. Although $\mathbf{F}_{\text{low}}^{\text{global}}$ can be directly employed for global semantic classification, it is not applicable in the two-layered approach.

C. Local Semantic Learning

SVM is employed as local semantic classifier in the first layer. It gives a good binary classifier that is used to find the decision function of optimal linear hyper-plane given labeled training data. The hyper-plane is linearly separable in the feature space (h). The input feature in the feature space (\mathbf{F}) is mapped onto the feature space via a nonlinear mapping ($\phi: \mathbf{F} \rightarrow h$), allowing one to perform nonlinear analysis of the input data using a linear method. In general SVM, a kernel is designed to map the input f space to the feature space. With the "kernel trick" property [11], [12], the kernel can be regarded as a similarity

measure between the two feature vectors without explicit computation of the map ϕ . Using the kernel function, the SVM classifier can be trained with training feature data. For this, an optimal hyper-plane is found to correctly classify the training data. By the optimization theorem of SVM, the decision function (Φ_n^{local}) can be formed to predict the local concept (x_n) of the input local feature vector ($\mathbf{F}^{\text{local}}$) as follows:

$$\Phi_n^{\text{local}}(\mathbf{F}^{\text{local}}) = \sum_t \{w_n(t) \cdot z_n(t) \cdot K(\mathbf{F}_n(t), \mathbf{F}^{\text{local}})\} + a_n \quad (3)$$

where K is kernel function that can be a linear, polynomial, or radial-basis function. And, $\mathbf{F}_n(t)$ is the t th support vector of the hyper-plane for x_n , w_n is the corresponding weighting vector of the support vector, a_n is the corresponding class vector of the support vector, and z_n is the threshold vector optimized for x_n .

In order to construct SVM classifier to produce a posterior probability, the output confidence of the SVM classifier is fitted to a parametric sigmoid model. The parametric sigmoid fitting model for the classifier of a local photo semantic x_n forms as follows:

$$P(x_n = 1 | \Phi_n^{\text{local}}(\mathbf{F}^{\text{local}})) \\ \cong \frac{1}{1 + \exp(A_n \cdot \Phi_n^{\text{local}}(\mathbf{F}^{\text{local}}) + B_n)} \quad (4)$$

where A_n and B_n are parameters to determine the shape of the sigmoid model for the local concept (x_n^{local}). So, the SVM output ranged from $-\infty$ to ∞ can be fitted to the probabilistic output ranged from 0 to 1.

The best parameter set of (A, B) is measured by solving the following regularized maximum likelihood problem with a set of labeled training examples. Given a training set $(\Phi_n^{\text{local}}(\mathbf{F}_i), x_i)$, let us define a new training set $(\Phi_n^{\text{local}}(\mathbf{F}_i), x'_i)$, where the x'_i is a target probability value. The new target value is used instead of $(-1, 1)$ for all of the training data in the sigmoid fit. This aims at making the new target value converge to $(0, 1)$ when the training set size approaches infinity. The new target value x'_i is defined as follows:

$$x'_i = \begin{cases} \frac{O_+ + 1}{O_+ + 2}, & x_i = 1 \\ \frac{O_- + 1}{O_- + 2}, & x_i = -1 \end{cases} \quad (5)$$

where O_+ is the number of positive samples and O_- is the number of negative samples. Then, the best parameter set for

a local photo semantic is obtained by minimizing the following cross-entropy error function

$$\arg \min_{(A, B)} \left[- \sum_i \{ x'_i \cdot \log p_i + (1 - x'_i) \cdot \log(1 - p_i) \} \right] \quad (6)$$

where p_i denotes $P(x_i | \Phi_n^{\text{local}}(F_i))$. We adopt an optimization method [17] to find the optimized parameters minimizing the above error function.

D. Local Semantic Classification

The first step for local semantic classification is to divide the whole region of an input photo into ten subregions by the PRT. Multiple low-level visual features are extracted from each local region and fed into the local concept detectors. For the local concept detection, let $\mathbf{R} = \{R_1, R_2, \dots, R_{10}\}$ be a set of the local regions. Then, the features of the local regions (R) are denoted as $\mathbf{F}^R = \{\mathbf{F}_{\text{cam}}^R, \mathbf{F}_{\text{low}}^R\}$. (2) and (3) can be specified for the local regions as follows:

$$\begin{aligned} P(x_n | \mathbf{F}^R) &= P(x_n | \mathbf{F}_{\text{cam}}^R, \mathbf{F}_{\text{low}}^R) \\ &= \frac{P(x_n) \cdot P(\mathbf{F}_{\text{cam}}^R | x_n) \cdot P(\mathbf{F}_{\text{low}}^R | x_n)}{P(\mathbf{F}_{\text{cam}}^R) \cdot P(\mathbf{F}_{\text{low}}^R)} \end{aligned} \quad (7)$$

where \mathbf{F}_{cam} is the same over all local regions in an input photo and $P(\mathbf{F}_{\text{low}}^R | x_n)$ is a likelihood rate of $\mathbf{F}_{\text{low}}^R$ about x_n . $P(\mathbf{F}_{\text{low}}^R | x_n)$ is estimated by a sigmoid model as follows:

$$P(x_n | \mathbf{F}_{\text{low}}^R) \cong \frac{1}{1 + \exp(A_n \cdot \Phi_n^{\text{local}}(\mathbf{F}_{\text{low}}^R) + B_n)} \quad (8)$$

Likewise, $P(\mathbf{F}_{\text{cam}}^R | x_n)$ is a likelihood rate of $\mathbf{F}_{\text{cam}}^R$ about x_n . So, it is also estimated by a sigmoid model as follows:

$$P(x_n | \mathbf{F}_{\text{cam}}^R) \cong \frac{1}{1 + \exp(A_n \cdot \Phi_n^{\text{local}}(\mathbf{F}_{\text{cam}}^R) + B_n)} \quad (9)$$

The probability set of x_n over all local regions (R), denoted as \mathbf{P}_n , can be written as follows:

$$\begin{aligned} \mathbf{P}_n &= \{P(x_n | \mathbf{F}^{R_1}), P(x_n | \mathbf{F}^{R_2}), \dots, P(x_n | \mathbf{F}^{R_{10}})\} \\ &= \bigcup_{i=1}^{10} P(x_n | \mathbf{F}^{R_i}). \end{aligned} \quad (10)$$

The probability set about the local concept set (X), denoted as \mathbf{P}_X , can be written again as follows:

$$\mathbf{P}_X = \bigcup_{n=1}^N \mathbf{P}_n = \bigcup_{n=1}^N \bigcup_{i=1}^{10} P(x_n | \mathbf{F}^{R_i}). \quad (11)$$

Then, given a local region, a local semantic vector with the probability values is composed for each local concept. The local semantic vector presents the likelihood of local semantics on

local photo regions. If v_n^R is replaced by $P(x_n | \mathbf{F}^R)$, (11) can be written again as follows:

$$\mathbf{V} = \{v_1^1, \dots, v_N^1, v_1^2, \dots, v_N^2, \dots, v_1^{10}, \dots, v_N^{10}\} = \bigcup_{n=1}^N \bigcup_{i=1}^{10} v_n^i. \quad (12)$$

IV. GLOBAL SEMANTIC CLASSIFICATION

A. Regional Semantic Merging

To find more likelihood semantics on the overlapping local regions, a concept merging is performed. It operates based on a semantic confidence map (SCM). The SCM is used to keep the most confident concept for basis local region set, denoted as $\mathbf{R}^{\text{basis}}$. The $\mathbf{R}^{\text{basis}}$ consist of one center and four corner regions, i.e., $\{R_1, R_2, R_3, R_4, R_5\}$, where rightly $\mathbf{R}^{\text{basis}} \subset \mathbf{R}$. The SCM is a set of five groups of overlapping local regions as shown in Fig. 5. SCM 1, denoted as $\mathbf{R}_1^{\text{map}}$, includes R_1 and R_{10} regions. SCM 2, denoted as $\mathbf{R}_2^{\text{map}}$, includes R_2, R_6, R_8 , and R_{10} regions. SCM 3, denoted as $\mathbf{R}_3^{\text{map}}$, includes R_3, R_6, R_9 , and R_{10} regions. SCM 4, denoted as $\mathbf{R}_4^{\text{map}}$, includes R_4, R_7, R_8 , and R_{10} regions. SCM 5, denoted as $\mathbf{R}_5^{\text{map}}$, includes R_5, R_7, R_9 , and R_{10} regions. Then, the confidence value of a local concept (x_n) of the a basis region ($R_b \in \mathbf{R}^{\text{basis}}$) is calculated as follows:

$$v_n^b = \max(v_n^R | R \in \mathbf{R}_b^{\text{map}}) \quad (13)$$

where, $v_n^2 = \max(v_n^R | R \in \{R_2, R_6, R_8, R_{10}\})$ in case of R_2 .

Given a basis region $R_b \in \mathbf{R}^{\text{basis}}$, (13) can be written again as follows:

$$\mathbf{V}^b = \{v_1^b, v_2^b, \dots, v_N^b\}. \quad (14)$$

B. Association of Local Semantics With Global Semantics

Global concept is a relatively higher level of semantics than all local concepts. Fig. 6 shows an example of the relationship between local and global photo semantics. There are eight local concepts (sky, tree-wood, flower, rock, bridge, windows, streets and building) and two global concepts (terrain and architecture). In sense of visual semantics, terrain is strongly linked with sky, tree-wood, flower, and rock while it is loosely linked with street bridge, windows, and buildings. On the contrary, architecture is strongly linked with bridge, windows, street, and building while it is loosely linked with sky, tree-wood, flower and rock.

From this observation, we consider the degree of strength of the semantic link between local and global semantic concepts. The higher value presents a stronger connection between local and global concepts. This approach could bridge the semantic gap between low-level feature and high-level semantic concept. Let $\mathbf{Y} = \{y_1, y_2, \dots, y_M\}$ be a set of M global concepts, subject to $M < N$. Global concepts are trained based on the likelihood of local SVM classifiers. The decision function (Φ_m)

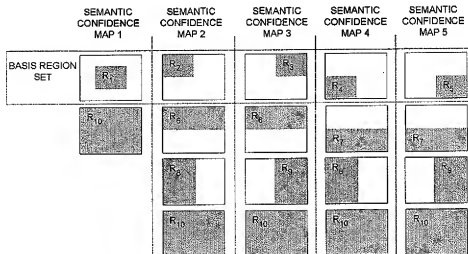


Fig. 5. Semantic confidence map.

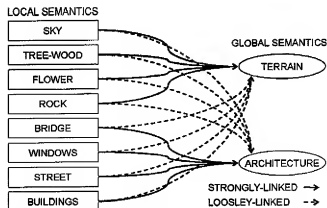


Fig. 6. Example showing the relationship between local semantics and global semantics.

to predict a global concept (y_m) of input local semantic vector (V^b) given a basis local region (R_b) is formed as follows:

$$\Phi_m^{\text{global}}(V^b) = \sum_s \omega_m(s) \cdot \zeta_m(s) \cdot K(V_m(s), V^b) + \alpha_m \quad (15)$$

where $V_m(s)$ is s th support vector of the hyper-plane for y_m , ω_m is the corresponding weighting vector of the support vector, ζ_m is the corresponding class vector of the support vector, and α_m is the threshold vector optimized for y_m .

The V^b constructs a semantic feature and it is fed into the global concept classifier. Like local semantic classifier, the joint conditional probability of a global concept $y_m \in Y$ given the semantic feature V^b vector for a photo region can be estimated by sigmoid fitting as follows:

$$P(y_m|V^b) = P(y_m|v_1^b, v_2^b, \dots, v_N^b) \propto \frac{1}{1 + \exp(A_m \cdot \Phi_m^{\text{global}}(V^b) + B_m)} \quad (16)$$

where A_m and B_m are parameters to determine the shape of the sigmoid model for the global concept (y_m).

C. Global Semantic Classification

Given a basis local region set, the merged confidence values for all local concepts are used to classify the local regions into the target classes. Since one of the main objectives is to detect multilabel detection meaning that an input photo can be labeled by one or more classes, we propose a criterion for multilabel categorization. Potential categories of the photo are determined given the probability values for the five basis local regions of an input photo. First of all, the probability values for all basis local regions are aligned in ascending order.

Then, the top- K classes with respect to probability value are assigned to classes of the input photo, whose probability values should be close enough, i.e., higher than a threshold. The classifier assigns the class of an input photo to multiple global concepts that should be satisfying the following condition, given by

$$c_m = m, \text{ if } P(y_m) \cdot P(y_m|V^b) \geq P_{th} \quad (17)$$

for any class and any basis region

where c_m is the predicted class of the input photo and P_{th} is the heuristic threshold value for the categorization.

V. EXPERIMENTS

Experiments were performed with 3086 photos that comprise the official test database of the MPEG-7 visual core experiment 2 (VCE-2). The VCE-2 aims to achieve photo categorization by using MPEG-7 visual descriptors. It also offers corresponding ground truth (GT) set, which has been cross-verified by several VCE-2 participants who had taken them. The GT set is given by seven semantic categories popularly appeared in home photos: "architecture," "indoor," "waterside," "nightscape," "snowscape," "terrace" and "sunset." In order to cover any preference of human visual perception, the GT set has been strictly made in detail. That is, an important rule in the GT decision was that a photo could be assigned with any semantic concept of which

TABLE I
STATISTICS OF THE TEST PHOTOS

Category	p_1^a	p_2^b	$L\text{-ratio} (=p_1/p_2)$
Architecture	1280	2370	1.85
Indoor	803	1047	1.30
Terrain	1297	2499	1.93
Night	372	758	2.04
Snowscape	175	336	1.92
Sunset	70	193	2.76
Waterside	650	1426	2.19

^a p_1 is the number of true photos that belong to the category defined in the left-most column; ^b p_2 is the number of true categories of the photos defined in the second column.

a scene could be found even in very small portion of photo. Thus, any of the test photos was often labeled by multiple true classes. It should be noted that the strictly-made GT set may cause some degradation in categorization performance.

Table I shows the statistics of the test photos. The test photos mostly belong to architecture or terrain category since they were usually taken during ordinary events. A photo exhibits about two categories on average and about four on maximum. We measured how many other categories appear with a certain category at the same time. To numerically quantify this relation, we exploited a criterion, called L -ratio, where L stands for local. For a category, the L -ratio is calculated by a relative rate of the number of true photos belonging to a certain category over the number of true categories of the photos. So, the bigger L -ratio of a category is, the more categories appear in the test photos at the same time. As can be seen in Table I, indoor and sunset showed the lowest and highest L -ratio, respectively. This means that indoor would often appear alone on the test photos while sunset would often appear with other categories. For example, as seen in Table I, there are 1280 photos that belong to the architecture. The 1280 photos contain 2370 true categories including architecture and any others.

In the experiment, 1597 photos were used for training. They were also from the MPEG-7 VCE-2 official training data, which is totally independent of the test data set. Of the training photos, 800 photos were from general home photos and 797 photos were from the Corel photo collection. To construct a set of local concept, we built nine important families of concepts that would frequently appear in local regions of home photos. The families of the local concepts consist of "ground," "human," "indoor," "mountain," "night," "plants," "sky," "structure," and "water." The concept families are subcategorized to the 34 local concepts as follows: "ground" to "gravel," "park," "pavement," "road," "rock," "sand," and "sidewalk;" "human" to "face" and "people;" "indoor" to "indoor" and "indoor-light;" "mountain" to "field," "peak," and "wood;" "night" to "night" and "street-light;" "plants" to "flowers," "leaves," and "trees;" "sky" to "cloudy," "sunny," "sunset," and "sunset-on-mountain;" "structure" to "brick," "arch," "building," "wall," and "windows;" "water" to "high-wave," "low-wave," "still water," "mirrored water," "beach," and "ice (snow)."

For training local concepts, we patched the training photo to local regions and then selected positive and negative samples by human visual perception. Then, multilevel visual features were extracted from the patched photo database. Several color

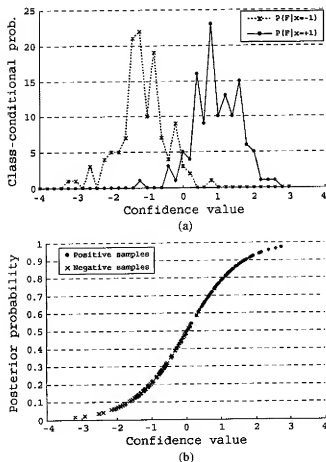


Fig. 7. Example of sigmoid model. (a) Histogram of the training samples for indoor classifier. (b) Best fit sigmoid for the indoor classifier (where $A_n = -1.2983$ and $B_n = 0.0268$).

and texture descriptors of the MPEG-7 were considered for the feature extraction [26], [27]: color structure (CS), color layout (CL), and scalable color (SC) descriptors are used for color features; and homogeneous texture (HT) and edge histogram (EH) descriptors are used for texture features.

The sigmoid model parameters were calculated for each local semantic classifier. Fig. 7(a) shows the histogram of positive and negative samples for indoor classifier. The solid line is the probability of negative samples while the dashed line is that of positive samples. As shown in Fig. 7(a), the histogram is not Gaussian, probably due to the small number of training data. Fig. 7(b) is derived by using Bayes' rule on the histogram estimates of the class-conditional densities. As can be seen, the sigmoid fit works well with relatively small number of training samples.

We measured performance with accuracy and precision since the numbers of the test data are imbalanced over all classes. As in general definition, $\text{accuracy} = (TP + TN)/(\text{total number of samples})$, and $\text{precision} = TP/(TP + FP)$, where TP, TN, FP, and FN stand for "true positive" when the case is positive and predicted positive, "true negative" when the case is negative and predicted negative, "false positive" when the case is negative but predicted positive,

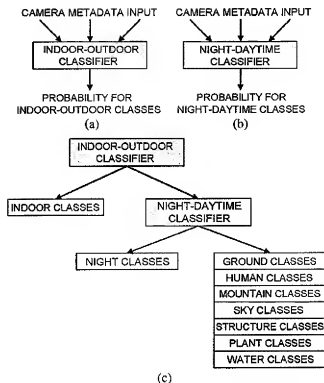


Fig. 8. Local semantic classifier for the camera metadata. (a) Indoor-outdoor classifier. (b) Night-daytime classifier. (c) Combination of the two classifiers to detect local photo semantics.

and "false negative" when the case is positive but predicted negative, respectively.

For the camera metadata, we selected exposure-time (ET), aperture number (AN), focal length (FL), and flash-fired or not (FF). As seen in the Boutell's studies, useful camera metadata is related to scene brightness. In our experiment, we also see that the camera metadata is useful only in a limited set of categories classified by scene brightness. However, many local semantics are not directly related to scene brightness. In order to extend the use of the camera metadata to all potential local semantic classifiers, we divide the local classifiers into night/daytime and indoor/outdoor classifier groups. From this way, the estimated posterior probability of the camera metadata can be assigned for all local classifiers. Fig. 8 shows the local semantic classifiers for camera metadata. Fig. 8(a) shows indoor/outdoor classifier that outputs probability values about indoor and outdoor classes. Likewise, Fig. 8(b) shows night/daytime classifier that outputs probability values about night and daytime classes. To associate the two camera metadata classifiers with the 34 local concepts, we make a classification scheme as seen in Fig. 8(c). As such, the first step is to classify the input camera metadata into indoor or outdoor classes. The indoor probability is assigned to indoor classes while the outdoor probability is assigned to outdoor classes. The second step is to classify the input camera metadata into night and daytime classes. The night probability is assigned to all night classes, and the daytime probability is assigned to all daytime classes including ground, human, mountain, plant, sky, structure, and water classes.

TABLE II
PERFORMANCE WITH ONLY CAMERA METADATA

Combination of camera metadata	Average performance ^a	
	Indoor/outdoor categorization (%)	Night/daytime categorization (%)
ET	73.07	72.98
AN	74.63	73.20
FL	26.09	16.77
FF	74.40	82.07
ET + FF	74.60	81.92
ET + AN	81.89	75.34
ET + AN + FF	82.57	78.38

^a Average performance is measured by averaging recall and accuracy.

For the purpose of choosing the best combination of camera metadata, categorizations with different combination of camera metadata were performed in indoor/outdoor and night/daytime classes. Before measuring the categorization performance, we also measured Kullback-Leibler (KL) divergence [28], [29], as studied in [25]. In our result, the combination of the ET, AN, and FF also showed the greatest KL divergence, whose values were about 2.30 and 1.96 in indoor/outdoor and night/daytime classification, respectively.

Table II shows categorization performance with only camera metadata. As expected from the KL-divergence measured above, ET, AN, and FF gives the best combination of camera metadata in both indoor/outdoor and night/daytime classes.

To verify the proposed combination of camera metadata and local visual semantics, experiments were performed with three different cases: 1) categorization by using only global low-level features; 2) categorization by using a combination of regional low-level features and local semantic features on the proposed two-layered scheme; and 3) categorization by the additional use of syntactic camera metadata on top of the second case. Table III shows a comparison of categorization performances by the three cases. As predicted in our study, overall performance is low in the case 1). Although some global categories, such as night, sunset, and snowscape showed relatively higher performance, other local categories did not. That is, global semantics would be captured better than local semantics. Using a combination of regional low-level features and local semantic features leads to more than 5% increase of the performance. In this case, the proposed scheme was adopted, but there was no use of camera metadata. This means that the proposed local semantic features provide more useful cues for local photo semantic classification. Finally, when we used syntactic camera metadata on top of the second case, the performance was more enhanced. There was about 5% increase in architecture category, about 7% increase in indoor category, about 5% increase in terrain category, about 2% increase in night category, about 1% increase in snowscape category, about 2% increase in sunset, and about 2% increase in waterside category. Fig. 9 shows categorization performance of the proposed method over various thresholds.

The proposed method was compared to a major referenced work using Bayesian network classifier based on global visual features and camera metadata [25]. The main difference of the proposed method from the Boutell's one is that we provide a multilayered scheme to model local and global semantics for

TABLE III
PERFORMANCE COMPARISON OF CATEGORIZATION SCHEMES WITH DIFFERENT FEATURES

Category	Recall / accuracy (%)		
	(a) With only global low-level features	(b) With only local semantic features	(c) With local semantic features & camera metadata
Architecture	61.01 / 61.08	66.43 / 67.82	70.13 / 72.23
Indoor	63.98 / 64.21	75.34 / 74.93	83.59 / 82.33
Terrain	67.52 / 67.60	74.42 / 72.73	80.24 / 77.48
Night	89.53 / 89.71	91.57 / 89.85	94.38 / 92.32
Snowscape	75.76 / 74.84	82.23 / 83.70	83.10 / 84.36
Sunset	76.47 / 77.41	78.77 / 81.29	81.67 / 82.34
Waterside	67.64 / 68.02	72.16 / 71.34	74.32 / 72.72

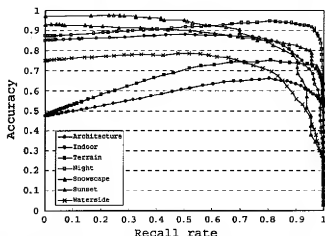


Fig. 9. Recall-accuracy curve of the categorization with local semantic and camera metadata.

multilabel detection. Boutell's method, in [40], was concerned in that it would be weak in multilabel classification. Thus, our presumption is that the proposed method would outperform the conventional one, especially Boutell's method, in multilabel classification if the advantages of the proposed two-layered SVM classifiers have been realized as it should. Table IV shows categorization performance of the two different methods. The training and testing data was the same as the above experiment. As seen in the result, almost categories except the architecture category were better detected by the proposed method than by the conventional method. In indoor and terrain, both methods showed similar performance. But, the proposed method much better detected other categories such as night, snowscape, sunset and waterside. As shown above, architecture, terrain, and indoor categories showed low L -ratio, i.e., the categories would often appear alone. On the contrary, night, snowscape, waterside, and sunset categories showed relatively high L -ratio, i.e., they would often appear with other categories. It means that the proposed method would be useful to detect multiple categories of generic home photos, thus providing more general applications to photo categorization.

For multilabel detection, we compared categorization error rates according to the number of true categories per photo. Fig. 10 shows the categorization error rates in the proposed method using two-layered SVM, where the number of true

TABLE IV
PERFORMANCE COMPARISON WITH BAYESIAN NETWORK SCHEME

Category	Bayesian network			Proposed two-layered SVM		
	Recall	Accuracy	Average	Recall	Accuracy	Average
Architecture	87.34	70.97	79.16	70.13	72.23	71.18
Indoor	96.64	71.93	84.29	83.59	82.33	82.96
Terrain	90.28	66.32	78.30	80.24	77.48	78.86
Night	84.14	67.79	75.97	94.38	92.32	93.35
Snowscape	87.43	41.95	64.69	83.10	84.36	83.83
Sunset	84.29	58.59	71.44	81.67	82.34	82.11
Waterside	73.08	55.41	64.25	74.32	72.72	73.52

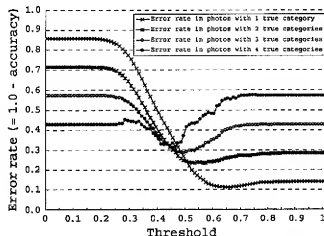


Fig. 10. Categorization error rate with respect to the number of true categories appeared in a photo.

categories per photo is ranged from 1 to 4. The error rate was calculated by (1.0-accuracy). Since the outputs of the SVM classifiers were fitted to the associated sigmoid model, the threshold is ranged from 0.0 to 1.0. In most cases, the minima of the error rates can be found around 0.5. This means that the SVM classifier has been well-trained to generate a probabilistic output. As seen in Table V, we compared the proposed method to Boutell's one in the categorization error rate. Since Boutell's method uses the *maximum a posteriori* (MAP) criterion, it probably gives only one categorization result. In the case of the proposed method, the error rate was about 11% when photos exhibit only single visual category. Even though the error rate increased when the number of true categories per photo increased, the amount of the increase of the error rate gradually decreased, e.g., the error rate increased by about 3% when the number of true categories increased to 4 from 3. In the case of Bayesian network, the error rate increased over all numbers of true categories per photo, compared to that in the case of the proposed method. When the number of true categories per photo increased, the error rate of the Bayesian network method was more degraded compared to that of the proposed method. This is because Boutell's method combined metadata with global low-level features. We see that the proposed method would outperform the Boutell's method in the multiclass classification.

Finally, we also performed comparative experiments with a k-NN approach [41]. The k-NN approach was performed for the

TABLE V
COMPARISON OF CATEGORIZATION ERROR RATES ACCORDING TO THE NUMBER
OF TRUE CATEGORIES APPEARED IN A PHOTO

Number of true categories / photo	1	2	3	4
Minimum error rate (%) for the proposed two-layered SVM scheme (corresponding threshold at the minimum error rate)	11.30 (0.66)	23.40 (0.53)	28.44 (0.49)	31.69 (0.46)
Minimum error rate (%) for Bayesian network scheme	18.11	30.44	43.69	54.72

TABLE VI
PERFORMANCE COMPARISON WITH K-NEAREST (K-NN) SCHEME

Category	Performance of k-NN method (%)
Architecture	60
Indoor	52
Terrain	65
Night	65
Snowscape	51
Sunset	25
Waterside	35

MPEG-7 VCE-2. It also used five MPEG-7 visual descriptors (SC, CL, CS, HT, and EH), which is the same as our method. The difference of this method from ours is that it modeled k-NN classifiers with only global low-level visual features. The k-NN approach did not utilize any camera metadata and multilayered classification with local and global semantics. Table VI shows performance comparison of different learning and classification schemes. As shown in the result, the average performance of the k-NN method was much lower than that of the proposed method over all categories. This simply shows that SVM based on SRM outperforms than k-NN based on ERM in generic classification.

VI. DISCUSSIONS AND CONCLUSIONS

We exploit a photo categorization scheme that utilizes both camera metadata and semantic features in two-layered scheme. The camera metadata would provide useful cues independent of photo contents as combined with low-level visual features for the purpose of measuring the likelihood rate about local photo semantics. However, error accumulation is problematic when using many probability models. The error accumulation often results from feature diversity and high dimensionality, and thus it is expected in the proposed two-layered feed-forward SVM classifiers. However, the proposed two-layered scheme reduces the error accumulation since it performs each classification in totally different two feature space; one is low-level feature space and the other is local semantic probability space. The error exhibited in the first SVM layer can be also reduced in the second SVM layer since the second SVM layer considers only correlation between local photo semantics, which is estimated from low-level feature, rather than the low-level feature itself. In addition, it is possible that there is no metadata in some photos. However, it is not very common in newly-acquired photos from users. Our target application is a home photo album that enables semantic photo categorization without users' manual ef-

forts. In this situation, users may not intentionally remove the metadata because they probably want to enjoy that kind of enhanced album functionality in organizing their photos for convenience.

The efficacy of the proposed classification method was demonstrated with 3086 home photos from MPEG-7 VCE-2 official databases. Compared to a major conventional method using global low-level features and camera metadata modeled by a Bayesian network, the proposed two-layered SVM performed better to classify visual semantics for general home photos. However, we need to do in-depth and detailed analysis on the profits due to the multilayered approach with camera metadata for multilabel detection in future work. Furthermore, we need to employ other syntactic features, such as keyword, spatial context, user preference, etc., for the proposed photo categorization scheme, and optimize the features to enhance the current performance.

ACKNOWLEDGMENT

The authors would like to thank Dr. R. O'Callaghan and Dr. M. Bober, Mitsubishi Electronics, and Dr. A. Yamada, NEC Corporation, who participated in MPEG-7 VCE-2, for their helpful comments and advices.

REFERENCES

- [1] T. Yamazaki and D. Gengras, "Image classification using spectral and spatial information based on MRF models," *IEEE Trans. Image Process.*, vol. 4, no. 9, pp. 1333–1339, Sep. 1995.
- [2] J. R. Smith and C. S. Li, "Image classification and querying using composite region template," *J. Comp. Vis. Pattern Recognit.*, vol. 75, no. 1, pp. 165–174, 1999.
- [3] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1055–1064, May 1999.
- [4] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, 2000.
- [5] S. Yang, J. H. Yoon, H. K. Kang, and Y. M. Ro, "Category classification using multiple MPEG-7 descriptors," in *Proc. Int. Conf. Imaging Sci., Syst., Technol.*, 2002, vol. 1, pp. 396–401.
- [6] S. Yang, J. H. Yoon, and Y. M. Ro, "Automatic Image Categorization using MPEG-7 Description," *Proc. SPIE Electron. Imag. Internet Imag.*, vol. 2018, pp. 139–147, 2003.
- [7] J. Smith, M. Naphade, and A. Natshe, "Multimedia semantic indexing using model vectors," in *Proc. IEEE Int. Conf. Image Process.*, 2002, vol. 2, pp. 445–448.
- [8] S. Newsam, B. Sumengen, and B. S. Manjunath, "Category-based image retrieval," in *Proc. IEEE Int. Conf. Image Process.*, 2003, vol. 2, pp. 596–599.
- [9] S. Kim, S. Yang, K. S. Seo, Y. M. Ro, J. Kim, and Y. Seo, "Home photo categorization based on photographic region templates," in *LMCS*, 2005, vol. 3689, pp. 328–338.
- [10] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.
- [11] K. Müller, K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf et al., "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, 2001.
- [12] B. Schölkopf and A. Smola, *Learning With Kernels*. London, U.K.: MIT Press, 2002.
- [13] P. Murphy, A. Torralba, and W. Freeman, "Using the forest to see the trees: a graphical model relating features, objects and scenes," in *Neural Information Processing Systems*. Cambridge, U.K.: MIT Press, 2003.
- [14] E. Sudderth, A. Torralba, W. T. Freeman, and A. Wilsky, "Learning hierarchical models of scenes, objects, and parts," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, vol. 2, pp. 1331–1338.
- [15] M. Naphade and T. Huang, "A probabilistic framework for semantic video indexing, filtering and retrieval," *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 141–151, 2001.

- [16] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifier*. Cambridge, MA: MIT Press, 2000.
- [17] H. T. Lin, C. J. Lin, and R. C. Weng, "A note on Platt's probabilistic outputs for support vector machines Dept. Comp. Sci., National Taiwan Univ., 2003 [Online]. Available: <http://www.cse.ntu.edu.tw/~cjlin/papers/plattprob.ps>, Tech. Rep.
- [18] Exchangeable Image File Format for Digital Still Cameras, JEITA CP-3451, Japan Electronics and Information Technology Industries Association, 2002.
- [19] J. H. Lim, Q. Tian, and P. Muelmen, "Home photo content modeling for personalized event-based retrieval," *IEEE Trans. Multimedia*, vol. 10, no. 4, pp. 24–37, 2003.
- [20] A. C. Loui and A. Savakis, "Automated event clustering and quality screening of consumer pictures for digital albuming," *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 390–402, 2003.
- [21] J. C. Platt, M. Czerwinski, and B. A. Field, "PhotoTOC: Automatic clustering for browsing personal photographs," in *Proc. 4th IEEE Pacific Rim Conf.*, 2003, vol. 1, pp. 6–10.
- [22] M. Cooper, J. Foote, A. Gigensohn, and L. Wilcox, "Temporal event clustering for digital photo collections," in *Proc. ACM Multimedia*, 2003, pp. 364–373.
- [23] S. Yang, S. K. Kim, K. S. Seo, and Y. M. Ro, "Automated situation change clustering of home photos for digital albuming," in *Proc. SPIE Electron. Imag. Storage Retrieval*, 2005, vol. 5682, pp. 212–223.
- [24] S. Yang, K. S. Seo, Y. M. Ro, S. K. Kim, J. Kim, and Y. Seo, "User-centric digital home photo album," in *Proc. IEEE Int. Conf. Consum. Electron.*, 2005, pp. 226–229.
- [25] M. Boutell and J. Luo, "Beyond pixels: Exploiting camera metadata for photo classification," *Pattern Recognit.*, vol. 38, pp. 935–946, 2005.
- [26] Y. M. Ro and H. K. Kang, "Hierarchical rotational invariant similarity measurement for MPEG-7 homogeneous texture descriptor," *Electron. Lett.*, vol. 36, no. 15, pp. 1268–1270, 2000.
- [27] B. S. Manjunath et al., *Introduction to MPEG-7*. New York: Wiley, 2002.
- [28] Z. Sun, "Adaptation for multiple cue integration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2003, vol. 1, pp. 1440–1445.
- [29] C. Liu and H. Y. Shum, "Kullback-Leibler boosting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2003, pp. 587–594.
- [30] P. Salembier and F. Marqués, "Region-based representations of image and video: Segmentation tools for multimedia services," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1147–1169, Dec. 1999.
- [31] V. Orta, M. T. Ozu, P. J. Iglinski, B. Xu, and L. I. Cheng, "DISIMA: An object-oriented approach to developing an image database system," in *Proc. IEEE Int. Conf. Data Eng.*, 2000, pp. 672–673.
- [32] X. Shi and R. Manduchi, "A study on Bayes feature fusion for image classification," in *Proc. IEEE Workshop Statist. Alg. Comput. Vis.*, 2003, vol. 8, pp. 1–9.
- [33] G. Tsoumakas, I. Katakis, and I. Vlahavas, "A review of multi-label classification methods," in *Proc. 2nd ADIS Workshop Data Mining Knowl. Discov.*, 2006, pp. 99–109.
- [34] J.-M. Renders, E. Gaussier, C. Goutte, F. Pacull, and G. Csaruk, "Categorization in multiple category systems," in *Proc. 23rd Int. Conf. Mach. Learning*, 2006, pp. 745–752.
- [35] M. Boutell, X. Shen, J. Luo, and C. M. Brown, "Learning multilabel semantic scene classification," *Pattern Recognit.*, vol. 37, no. 9, pp. 1757–1771, Sep. 2004.
- [36] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor, "Kernel-based learning of hierarchical multilabel classification models," *J. Mach. Learning Res.*, vol. 7, pp. 1601–1626, 2006.
- [37] X. Li, L. Wang, and B. Sung, "Multilabel SVM active learning for image classification," in *Proc. IEEE Int. Conf. Image Process.*, 2004, pp. 2207–2210.

- [38] H. C. Kim, S. Pang, H.-M. Je, D. Kim, and S. Y. Bang, "Pattern classification using support vector machine ensemble," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2002, vol. 2, pp. 160–163.
- [39] Y. Yu, X. L. Wang, and B.-Q. Liu, "A gradual combining method for multiSVM classifiers based on distance estimation," in *Proc. IEEE Int. Conf. Mach. Learning Cybern.*, 2006, vol. 6, pp. 3434–3438.
- [40] M. Boutell, "Exploiting context for semantic scene classification," Univ. Rochester, Rochester, NY, Tech. Rep. 894, 2006.
- [41] R. O'Callaghan and M. Bober, *Results on Image Categorization (VCE-2)*, ISO/IEC JTC1/SC29/WG11 MPEG-7 Visual, M12049, Apr. 2005.



Seungji Yang received the B.S. degree from Kangwon National University, Kangwon, South Korea, in 2001 and the M.S. degree from the Information and Communications University (ICU), Daejeon, South Korea, in 2002, where he is currently working toward the Ph.D. degree.

His research interests include content adaptation, content-based image analysis, MPEG-7, and MPEG-21.

Dr. Yang received the 14th Outstanding Science and Technology Thesis Award sponsored by the Korean Federation of Science and Technology Societies in 2004.



Sang-Kyun Kim received the B.S., M.S., and Ph.D. degrees in computer science from the University of Iowa, Iowa City, in 1991, 1994, and 1997, respectively.

In 1997, he joined the Samsung Advanced Institute of Technology as a Research Staff Member. He is a member of Senior Research Staff as well as a Project Leader on "Image and Video Content Search" team in the Computing Technology Laboratory. His research interests include digital contents (i.e., image, video, and music) analysis and management, fast image search, and indexing, MPEG-7, and multimodal analysis. He serves as a Cochair of MPEG-7 Visual Core Experiment Group and a Project Editor of MPEG-7 Visual Group.



Yong Man Ro (M'92-SM'98) received the B.S. degree from Yonsei University, Seoul, South Korea in 1981 and the M.S. and Ph.D. degrees from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 1987 and 1992, respectively.

In 1987, he was a Researcher at Columbia University, and from 1992 to 1995, he was a Visiting Researcher in University of California at Irvine and KAIST. In 1996, he was a Research Fellow, University of California at Berkeley. In 1997, he joined Information and Communications University, Korea, where he is currently professor and director of Image Video System Laboratory. His research interests include image/video processing, MPEG-7, MPEG-21, Visual data mining, image/video indexing, and spectral analysis of image signal.

Dr. Ro received the Young Investigator Finalist Award in ISMRM in 1992. He is a member of SPIE and ISMRM. He has developed MPEG-7 texture descriptor and MPEG-21 DIA visual impairment descriptors and modality conversion.

Modulation and Coding for Information Storage

The success of certain modulation and coding techniques in data communications have inspired promising new applications in digital data storage.

by Paul H. Siegel and Jack K. Wolf

The communications industry is concerned with reliable and efficient means for transmitting information from one place to another. The information storage industry is concerned with reliable and efficient means for transporting information from one time to another. The teachings of information and communications theory apply equally well to both scenarios. In communications as well as storage systems, information must be transported through noisy channels that accept input signals (perhaps from some constrained class of signals) and produce output signals from which the transmitted or stored information must be recovered. Yet the readers of this magazine are, for the most part, unaware of the applications of their skills to the storage industry, which has worldwide annual sales in excess of \$50 billion. (This figure is for magnetic storage alone.)

While communicators are concerned with maximizing the rate (in bits per second) whereby digital information can be transmitted and reliably received, storage researchers are largely concerned with maximizing the areal density (in bits per square inch) and volumetric density (in bits per cubic inch) for storing and reliably retrieving information.

The storage industry has made steady progress increasing the density of information storage of digital data. Over the last 25 years, the areal density of storage of digital data on magnetic hard disks has grown geometrically at a compound growth rate of about 29%. This remarkable growth is reflected in Fig. 1, which shows the areal density of high-end IBM disk drives as a function of production year.

Most of this increased storage density has resulted from improvements in the part of the system that we call "the channel," including the storage medium itself, the read and write heads, mechanical features determining the flying heights and positioning of these heads, and so on.

Though less dominant, the contribution of advances in signal processing and coding has not been insubstantial. For example, channel modeling indicates that without the improvements in

run length-limited codes (described in more detail later) the compound growth rate in areal density would have been approximately 24 percent. If we restrict attention to linear density gains, the modeling shows that progress in signal processing and coding technology alone has accounted for almost a quadrupling of the linear density achievable with a "typical" set of recording components.

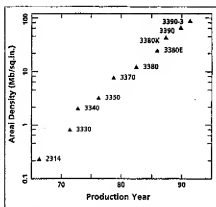
As communications theorists, we define the "channel" as the part of the system over which we have no control. As such, we assume that the channel is fixed. Although read/write engineers for storage systems may not have direct control over the choice of the channel, the systems that make up the channel are constantly being improved and these improvements have been the principle reason for the growth in information density in Fig. 1. Thus, instead of utilizing modern modulation and coding systems that yield performance closer to the channel capacity, the designers of these systems have taken the alternative (and probably harder) approach of increasing the channel capacity itself.

As a matter of fact, the modulation and coding systems used in most of today's products do not differ markedly from the modulation and coding systems used in products a decade ago. As we shall see, the modulation and coding systems in these products were chosen to match a particular type of detector, called a peak detector, which has been an integral part of the system. The peak detector has seen a remarkable life. However, we may be at a turning point, where the peak detector and its associated modulation/coding/signal processing systems could be replaced by a sampling detector and an entirely new type of modulation/coding/signal processing system. The new system looks very much like the systems that communications engineers are accustomed to seeing in advanced communications systems: partial-response equalization, trellis codes, Viterbi detection, adaptive digital filtering, and so on. Both the old and new systems will be described in this paper.

Although many different types of media can be utilized for the storage of digital data, we concentrate in this paper on systems based upon tra-

Paul H. Siegel is manager, Signal Processing and Coding project, IBM Almaden Research Center, San Jose, California.

Jack Kell Wolf is a chaired professor of Electrical Engineering and Computer Engineering and a member of the Center for Magnetic Recording Research at the University of California, San Diego.



■ Figure 1. Storage density versus time

ditional magnetic recording. The techniques described, however, have applications in other types of storage systems such as those using magnetic-optic and optical recording.

One important difference between communication systems and storage systems is the requirement on decoded error rate. Often in communication systems, the goal is a user error rate of 10^{-3} or 10^{-4} . Storage systems, however, often require error rates of 10^{-12} or better. Furthermore, implementations of error recovery procedures and their impact on the performance measures of storage devices have often mandated that there be a strict requirement on the "raw" error rate at the output of the channel before the error correction decoder.

Channel Model

The "guts" of a magnetic recording system are: the write head, the magnetic medium, and the read head. (The write head could be the same as the read head and usually has been for disk drives.) The write head is driven by a current source that carries the information to be stored. The write head radiates flux, which changes the state of magnetization of the magnetic medium immediately under the head. Actually, since the head is moving with respect to the magnetic medium, any point on the magnetic medium retains the state of magnetization corresponding to the last flux it experienced from the write head as the head moves away from that point.

On a rigid disk, the disk moves in a circular motion under the head. Information is stored on the disk in concentric tracks, the width of a track roughly being governed by the size of the write head. The density of recording is then the product of the number of tracks per inch (tpi) and the linear density of information along a track measured in bits per inch (bpi). Typical numbers for today's high end (i.e., expensive) rigid disk drives are: 3,000 tpi and 30,000 bpi.

There are at least two types of magnetic tape systems. In the first type, the head (or heads) remains stationary while the tape is pulled over the head. In the second, called a rotary head system, the head (or heads) is fastened to a spinning drum while the tape is moved slowly past the drum. This type of system is used in videocassette

recording and other applications where larger bandwidth is required. The head-to-tape speed in the first type is governed by the speed of the moving tape, while in the second type, the head-to-tape speed is mostly a function of the rotational speed of the drum and not the speed of the tape past the drum. In multiple write head tape systems, information usually is written simultaneously on many tracks, while in rigid disk storage systems, a single head almost always writes information on a single track. Rigid disk systems usually have a single head for both writing and reading (or, as in some recently announced products, write head physically coupled with a read head), while tape systems may have separate read head so that the system can read what is being written.

The current into the write head induces a magnetization pattern on the track immediately below the write head. When a track is to be read, a read head is positioned over the track. Then, the magnetization pattern "frozen" on that track radiates flux that is sensed, or "read," by the read head. The read head produces a voltage that is symptomatic of the magnetization on the track being read. There are primarily two types of read heads: inductive heads which contain coils of very fine wire and which produce a voltage proportional to the time derivative of the flux that passes through its coils, and magneto-resistive (MR) heads which produce a voltage directly proportional to the flux sensed by the head. MR heads produce larger read voltages than inductive heads, but have a limited dynamic range for linear operation. Only inductive heads have been used for writing, to this date.

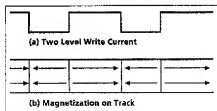
In rigid disk systems there is a separation between the head and the disk called the air bearing. This separation is extremely small and the slightest imperfection on the surface of the disk (or any contaminant in the air bearing) could cause the head to "crash." Tape systems have the head (or heads) in contact with the magnetic surface. Because of the roughness of the tape, there is actually a nonzero average separation between the head (or heads) and the tape.

The magnetic recording channel is inherently nonlinear because of the hysteresis that affects all magnetic media. However, magnetic recording specialists have found a way of linearizing the channel for a restricted range of inputs. They constrain the current into the write head to take on only two possible values, for example, $+A$ and $-A$, where the amplitude A is chosen sufficiently large so as to completely magnetize the magnetic storage medium in one of two directions. Thus, the hysteresis effect can be ignored. This type of recording is called saturation recording, and all practical digital storage devices use this approach. After information has been written on a track using saturation recording, the magnetic medium on that track would be alternately magnetized along the track, either in the direction of rotation or the opposite direction. A two-level write current waveform and its corresponding magnetization pattern on a track are shown schematically in Fig. 2.

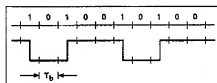
Since the write current can only take on the values $+A$ and $-A$, the stored information is represented by the times when transitions occur. Usually, one assumes that the time interval assigned to each channel bit is fixed. We call the

The magnetic recording channel is inherently nonlinear because of the hysteresis that affects all magnetic media.

A major problem affecting the read signal is the time-varying gain of the channel.



■ Figure 2. Saturation recording



■ Figure 3. NRZI recording

duration of a channel bit T_b . In one convention called NRZI modulation, a channel bit equal to "1" is written as a transition in the write current (from $-A$ to $+A$ or vice versa) in the middle of a bit cell, and a channel bit equal to 0 is written as no transition in the write current (i.e., the write current in the present bit cell remains at the same level as at the end of the previous bit cell). A representative sequence of channel bits and the corresponding write current after NRZI modulation are shown in Fig. 3.

Assume that the write current is initially equal to $+A$ and that transitions in the write current occur at times $j_1 T_b, j_2 T_b, \dots, j_i T_b$, where j_1, j_2, \dots, j_i are integers such that $j_1 < j_2 < \dots < j_i$. Let $u(t)$ be a unit step that occurs at $t = 0$,

$$u(t) = \begin{cases} 0 & \text{if } t < 0, \\ 1 & \text{if } t \geq 0. \end{cases}$$

Then the write current, $I(t)$, can be written as:

$$I(t) = A + 2A \sum_{i=1}^{\infty} (-1)^i u(t - j_i T_b).$$

Let $g(t)$ denote the read voltage corresponding to an isolated positive-going transition of the write current (from $-A$ to $+A$) occurring at time $t = 0$. Then, if the channel were linear, the output voltage, $V(t)$, corresponding to the magnetization induced by the write current $I(t)$ would be given as [1]:

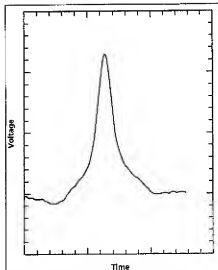
$$V(t) = A \sum_{i=1}^{\infty} (-1)^i g(t - j_i T_b).$$

A plot of a typical $g(t)$ taken from a thin film disk with thin film (inductive) heads is shown in Fig. 4.

A common mathematical model for such an isolated transition response is the Lorentzian pulse shape given by the formula:

$$g(t) = \frac{1}{1 + \left(\frac{2t}{PW_{50}}\right)^2}.$$

The constant denoted by PW_{50} is a parameter that represents the pulse width of the transition



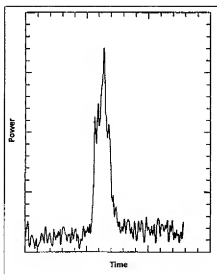
■ Figure 4. Isolated transition response from a thin film disk and thin film head

response at 50 percent of the maximum amplitude. The advantage of using this admittedly imperfect model is that it portrays the shape of the isolated transition response reasonably well, and requires only the single parameter PW_{50} .

The preceding discussion assumes the validity of using linear superposition of the isolated transition responses to form the composite output due to many transitions. This is, again, only a rough approximation to what actually occurs because of the interaction of the transition to be written with those that have already been written. The new transition could be very close to the previous one, and the magnetic flux "radiated" from the medium where the previous transition was written influences what currently is being written. This flux, called the demagnetization field, pulls the transition to be written closer to the previous transition. We will neglect nonlinear effects in this paper, but one must bear in mind that these cannot be completely ignored in the practical design of high density systems.

Another major problem affecting the read signal is the time-varying gain of the channel. This can be due to many phenomena, including variation in the spacing between the read head and the medium over time, or the presence of physical defects in the medium itself. In disk systems, a map of each disk surface is usually made, identifying portions of the disk containing defects so that these portions will not be used.

So far we have not discussed the effects of noise. It is common to assume that the noise in the system is additive and Gaussian. Usually, it is also assumed that the noise consists of two components: a Gaussian white noise component due to the electronics on the read (i.e., receiver) side, and a Gaussian colored component due to the medium. The spectral characteristics of this colored noise are essentially the same as would be obtained from passing white noise through the linear transfer function characterizing the system. More complicated models for the noise exist, particu-



■ Figure 5. Noise variance as a function of the position in a pulse

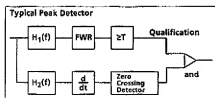
larly for thin film media, where it has been shown, for example, that there is more noise at the center of a transition than where there is no recorded transition. A plot of the variance of this noise as a function of position within an isolated transition response for a representative thin film medium is shown in Fig. 5. This signal-dependent noise leads to opportunities for improved detectors, but this advanced subject is still exploratory and lies beyond the scope of this paper.

Peak Detection Systems

The design of the modulation, coding and signal processing in past magnetic recording products has been driven by the detector chosen to detect the transitions in the channel input waveform. This detector, called a peak detector [2], has the advantage of being both robust and extremely simple to implement. However, by its very nature, it works best at low linear densities. A block diagram of a typical peak detector is shown in Figure 6.

There are two paths through the detector. The top path is used to "qualify" a peak, i.e., to ensure that the peak has sufficient amplitude. This path consists of a linear filter, $H_1(f)$, a full wave rectifier (FWR), and a threshold testing circuit. The bottom path is used to locate the peak by differentiating the signal after the linear filter $H_2(f)$ and then passing the differentiated signal through a zero-crossing detector. The detector only accepts a peak if the peak amplitude was large enough to pass the qualification test.

Once a peak is detected by the peak detector, it is thought to be due to a transition in the input waveform. A device called a phase-lock loop (PLL) is used to derive timing from the position of the detected peaks. The PLL produces a clock of period T_b seconds by which to identify channel bit intervals (sometimes called "bit cells"). Then, if an output pulse is located in a bit interval, that bit interval is said to contain a transition. Using



■ Figure 6. Block diagram of a peak detector

the NRZI precoding convention of Fig. 3, a bit interval with a transition corresponds to a recorded "1", and an interval without corresponds to a "0."

Note that the use of the NRZI precoder ensures that the reconstruction of the recorded sequence is insensitive to polarity inversion of the channel output waveform: a peak, regardless of its polarity, corresponds to a "1", and the absence of a peak corresponds to a "0." From another perspective, if the detector had to recover the actual recorded write current from the correspondence between peaks and write current transitions, a single error from a "missed" peak would propagate until the next "missed" peak.

The output of the peak detector is used as an input to the PLL, and the output clock produced by the PLL is constantly being adjusted so that the average peak position is centered with respect to the edges of the bit interval.

If one examines the waveform produced by the linear superposition of two Lorentzian pulses (of opposite sign) separated by αPW_{50} seconds, one finds that this waveform will contain two peaks separated by βPW_{50} seconds, where $\beta > \alpha$. The parameters α and β are related by the formula

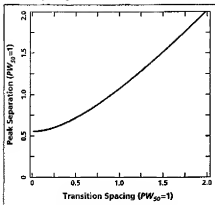
$$\beta = \frac{\alpha^2 - 1 + 2\sqrt{1 + \alpha^2 + \alpha^4}}{3}$$

which for small α becomes

$$\beta \approx \frac{1 + \alpha^2}{\sqrt{3}}$$

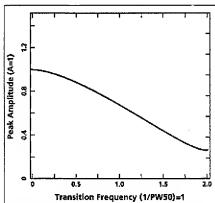
For α much greater than 1, β is approximately equal to α , but as α approaches zero, β approaches a fixed, limiting distance given by the value β_0 . Thus, the peaks will be centered in their bit interval only at low densities. Figure 7 shows a plot of

■ Figure 7. Peak separation as a function of transition spacing



Once a peak is detected by the peak detector, it is thought to be due to a transition in the input waveform.

A second performance enhancement technique for peak detection channels is the use of a modulation code.



■ Figure 8. Roll-off curve for Lorentzian response

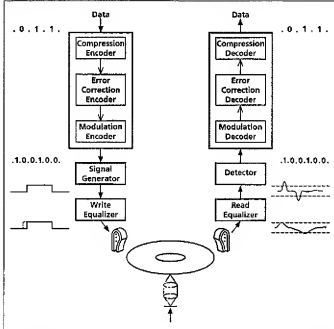
the normalized peak separation β as a function of normalized transition spacing α .

Of course, another effect of the interference of pulses is the decrease of the peak amplitude. The amplitude of peaks in the superposition of an infinite sequence of Lorentzian pulses (of alternating polarity) as a function of the transition separation $T = \alpha W_{50}$ seconds (sometimes called the "roll-off curve") also has a simple closed-form solution:

$$\gamma(\alpha) = \frac{(\pi/2\alpha)}{\sinh(\pi/2\alpha)}$$

where $\sinh(x) = (e^x - e^{-x})/2$ is the hyperbolic sine function. (For fans of complex analysis, we remark that one approach to deriving this formula provides a pleasant exercise in contour integration.) Fig-

■ Figure 9. Block diagram of a recording system with peak detection



ure 8 shows the rolloff curve as a function of normalized transition frequency. The amplitude $\gamma(\alpha)$ also plays a role in determining the design density as limited by the amplitude qualification of peaks in the detector.

To improve the performance of the system at moderate information densities, two techniques are used, often in combination. These techniques, "equalization" and "modulation coding," are now briefly described.

Equalization refers to the use of a linear filter at the output which changes the overall impulse response of the system before the peak detector. This equalizer is often called a pulse slimmer [3], its purpose being to create a channel whose isolated transition response is a thinner pulse than that produced by the unequalized channel. Many different equalization techniques are utilized, but they all have the effect of increasing the noise at the equalizer output since pulse slimming is achieved by boosting the high frequencies where there is little signal and much noise. We will not devote much space in this paper to the discussion of equalization. However, we will mention a new class of target transfer functions derived from partial-response signaling, a method familiar to many communications engineers.

A second performance enhancement technique for peak detection channels is the use of a modulation code, or, more specifically, a special class of these codes called run length-limited (RLL) or (d, k) codes. Here d and k are nonnegative integers with k strictly larger than d . A (d, k) -encoded sequence must satisfy the constraint that symbols "1" must be separated by at least d and at most k symbols "0." (The choice of the letters d and k to describe these codes is unfortunate because of their different meaning in the discussion of error-correction codes, but several attempts to change the notation have met with little success. Fortunately, in this paper we will make only brief remarks about error-correction codes, so no such confusion should arise.)

We will describe (d, k) codes in terms of the restrictions that these codes produce on the write current. Whereas, for the unencoded case, the minimum time interval between transitions in the write current can be as small as T_b and the maximum time interval between adjacent transitions could be infinite, when a (d, k) code is utilized, the minimum time interval between transitions is $(d+1)T_b$ and the maximum time interval between transitions is $(k+1)T_b$. Since the magnetic medium is moving past the head at some velocity, v , the minimum spacing between transitions on the medium is $v(d+1)T_b$. The value $v(d+1)T_b$ is referred to as the linear transition density, and if this distance is measured in inches, the linear density is measured in flux changes per inch (fci). Typical high- and medium-density magnetic disk systems today operate at between 20,000 and 30,000 fci, and magnetic tape systems operate at linear densities as high as 62,000 fci.

The necessity for the parameter k to be finite follows from the fact that the PLL requires feedback from the peak detector in order to position the bit interval boundaries. If there is a large time interval between transitions, there will be a corresponding large time interval during which the PLL sees no corrective signal. Such a situation would result in an undesirable drift in the clock signal produced by the PLL. One might think that a data

scrambler could be used to make the likelihood of such an event very small. However, disk designers are interested in worst case behavior: once a file written on a disk can no longer be read reliably, there is no second chance to retransmit the file.

A block diagram of a typical recording system using a peak detector, linear equalization, a (d, k) modulation code, and an error-correcting code is shown in Fig. 9.

The user data (possibly compressed) is first passed through an encoder for an error-correction code (ECC) such as for a Reed-Solomon code. ([4].) The output of the ECC encoder is then encoded again using a (d, k) modulation encoder. Finally, the output of the (d, k) encoder is NRZI modulated by the signal generator, forming a write current of two levels, where a transition in the write current corresponds to a "1" in the (d, k) -encoded stream. On the read side, the output voltage waveform from the read head is equalized and passed through a peak detector (the ever-present phase-lock loop is not shown), where bit cells containing detected peaks are converted to 1s and bit cells without peaks are converted to 0s. The corresponding binary stream from the peak detector is next passed through the (d, k) decoder, then through the ECC decoder (and decompressed, if appropriate). Notice that the NRZI precoding convention provides a very elegant way to translate the constraints imposed upon the readback signal by the peak detection system into constraints on the sequences to be generated by the modulation encoder.

Most communication engineers have some familiarity with data compression and ECC schemes, but not as much with modulation coding techniques, particularly those used in data storage. The remainder of this paper will be devoted to a discussion of such modulation codes, beginning with (d, k) coding.

(d, k) Codes

In this section we describe some of the (d, k) codes that have been used in commercial storage products. Further details may be found in a variety of references [5-8]. We begin with a very simple example of a (d, k) code where $d = 0$ and $k = 2$. This code, sometimes called Group Code Recording (GCR) [9], has found application in magnetic tape drives. The code description is shown in Table 1.

User bits are encoded, four bits at a time, into five-bit code words. It is easily verified that any concatenation of these five-bit code words satisfies the $(0, 2)$ constraint:

The rate of the GCR code is $4/5$ and one might wonder what is the maximum code rate possible for any $(0, 2)$ code. The answer to four decimal places is .8792. For (d, k) codes in general, the maximum code rate (sometimes called the capacity of the code) is given by the base-2 logarithm of the largest real root of one of the following equations, depending on whether k is finite or not:

$$x^{k+2} - x^{k+1} - x^{k-1} - \dots - 1 = 0, \quad k < \infty$$

$$x^{d+1} - x^d - 1 = 0, \quad k = \infty$$

A listing of the maximum code rates for some (d, k) codes is given in Table 2.

User Bits	Channel Bits
0000	11001
0001	11011
0010	10010
0011	10011
0100	11101
0101	10101
0110	10110
0111	10111
1000	11010
1001	11011
1010	01010
1011	01011
1100	11110
1101	01101
1110	01110
1111	01111

■ Table 1. Encoding/decoding table for $(0, 2)$ GCR code

The purpose for choosing the parameter d to be strictly greater than zero is to increase the information density along a track while keeping the time interval between adjacent transitions greater than some fixed constant. Assume that T_{min} is the smallest time interval that can be allowed between neighboring transitions in the two-level write current. If d is equal to zero, then we must choose T_b such that $T_b \geq T_{min}$. For this choice of T_b since one coded binary digit cannot contain more than one information bit, the maximum information rate that can be supported by the two-level input waveform would be $1/T_b \leq 1/T_{min}$ bits/second. Now assume that d is chosen as an integer strictly greater than zero. Since the minimum time interval between transitions is now $(d+1)T_b$, T_b can be chosen to equal $T_{min}/(d+1)$ and the coded binary digits then occur at a rate of $1/T_b = (d+1)/T_{min}$ binary digits/second. For a fixed value of T_{min} , this corresponds to a coded symbol rate that is a factor of $(d+1)$ times the transition rate. Unfortunately, this increase is not in the information rate but in the coded binary symbol rate, and the number of coded binary digits required to represent one information bit generally increases as d does. Let R be the ratio of the number of information bits to coded binary digits for a given (d, k) code. Then the information rate for this system using the (d, k) code is $R/T_b = R(d+1)/T_{min}$ bits/second. The product $R(d+1)$, called the density ratio of the code [10], represents the increase (if $R(d+1) > 1$) or decrease (if $R(d+1) < 1$) in information rate using a (d, k) code as compared to an uncoded system. We are particularly interested in systems where the density ratio is strictly greater than 1, for then we are storing information at a higher

■ Table 2. Maximum code rates for selected (d, k) constraints

	d=0	d=1	d=2
k=1	.6942		
k=2	.8792	.4057	
k=3	.9468	.5515	.2878
k=4	.9752	.6175	.4057
k=5	.9881	.6509	.4650
k=6	.9942	.6690	.4979
k=7	.9971	.6793	.5174

The (1,3)

code has

many names:

Miller code,
Delay Modulation, and
Modified
Frequency
Modulation
(MFM).

d	k	m	n	R(d+1)
0	2	4	5	.8
1	3	1	2	1.0
1	7	2	3	1.33
2	7	1	2	1.5

Table 3. Parameters of commonly used (d,k) codes

rate than the highest possible transition rate allowed in the two-level input waveform. The most common (d,k) codes used in past and present products are listed in Table 3.

We have already discussed the (0,2) (GCR) code. The (1,3) code has many names, including Miller code, Delay Modulation, and Modified Frequency Modulation (MFM) code [9]. It is a rate 1/2 code where one information binary digit is encoded into two coded binary digits. It is a systematic code in that the information sequence $i_1, i_2, \dots, i_k, \dots$ gets transformed into the coded sequence $i_1, i_1, i_2, i_2, \dots, i_k, i_k, \dots$. There is a simple rule for inserting the extra digits (z_k): Choose z_k to be 0 unless it is to be inserted between two information 0s, in which case choose z_k to be a 1.

We will now demonstrate how to derive the coding rule for the (1,3) code. The derivation illustrates some of the basic ideas involved in the design of constrained recording codes. However, more powerful techniques are required in general, as will be discussed in more detail. We begin with a finite state transition diagram (FSTD) which produces all binary sequences satisfying the (1,3) constraint. This FSTD is shown in Fig. 10. Constrained code sequences are generated by taking walks on the graph, following the arrows and reading off the code symbols that label the edges traversed.

The capacity for a (1,3) code is $C = .5515$, and we desire a code rate close to C , such as a rate $R = 1/2$. Thus, we seek a graph that can represent a rate 1/2 encoder finite-state-machine, namely, one with two edges emanating from each state (one for encoding a 0, the other for encoding a 1) and with binary code words of length 2 on each edge. In order to obtain a graph with edges labeled by code sequences of length 2, we form the "second power" of the FSTD of Fig. 10, that is, the FSTD obtained by taking steps of size 2 in Fig. 10. This graph is shown in Fig. 11.

Note that state 3 in this graph is deficient in that it has only one edge emanating from it. However, we are very fortunate in that the only way to enter state 3 is from state 1, and state 1 has 3 edges emanating from it. Eliminating the edge that goes from state 1 to state 3 eliminates state 3 from consideration altogether, resulting in the FSTD shown in Fig. 12.

We now note that states 1 and 2 are "equivalent," meaning that both states produce the same set of code sequences. Thus, these states can be combined into one state as shown in Fig. 13 [8].

Finally, we obtain the state diagram for a rate $R = 1/2$ encoder for a (1,3) code by labeling the edges in the form abc , where a is the information bit that is the input to the encoder and bc is the pair of (1,3) constrained binary digits produced by the encoder. This is shown in Fig. 14. This encoder is precisely the same encoder as used in the Miller (1,3) code.

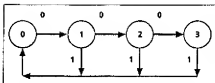


Figure 10. Finite-state transition diagram for (1,3) sequences

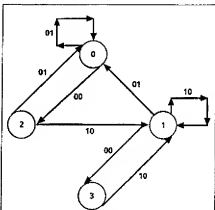


Figure 11. Two-step transition diagram for (1,3) constrained binary sequences

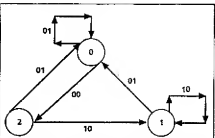


Figure 12. Modified two-step transition diagram for (1,3) constrained binary sequences

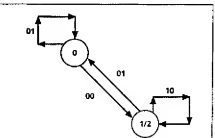
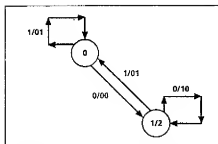


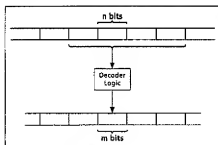
Figure 13. Two-state transition diagram for rate 1/2 (1,3) constrained sequences

The rate 2/3 (1,7) code is arguably the most popular (d,k) code in use today. Several variations of this code exist [11-13]. A simple and elegant description, due to Jacoby, begins with the encoding table in Table 4.

If the encoder is based solely upon a table lookup, however, we note that the user data sequences 00,00,



■ Figure 14. Two-state encoder for rate 1/2 (1,3) code



■ Figure 15. Sliding block decoder

00.01, 10.00, and 10.01 produce channel sequences (namely, 101.101, 101.100, 001.101, and 001.100, respectively) which violate the (1,7) constraint. The "substitution table" in Table 5 is used to correct for these violations.

When the encoder is ready to encode a pair of user data bits, it "looks ahead" to the next pair of user data bits to see if using Table 4 for both pairs would result in a violation of the (1,7) constraint. If no violation would occur, the encoder uses Table 4 to encode the first pair. If a violation would occur, the encoder encodes these two pairs using Table 5.

Decoding can be accomplished in a state-dependent manner using a "sliding-block decoder," a generalization of the code word "table look-up" decoder used for the (0,2) and the (1,3). A schematic of a sliding-block decoder is shown in Fig. 15.

As illustrated for a rate m/n code, the decoding of a code word of length n depends on the contents of a decoder window that contains the code word in question, as well as a fixed number of past and future code words ("look-back" and "look-ahead"). In the case of the (1,7) code, the decoder decodes the current three-bit code word by looking ahead at the next two upcoming code words. In this way, a single incorrectly detected code symbol can propagate into a burst of at most six user bits (in fact, the burst length does not exceed five user bits).

Another modulation code used in several disk drive products is a rate 1/2 (2,7) code. One encoding and decoding table for such a code, invented by P. Franzese, is given in Table 6.

It is easily seen that the code rate is 1/2 (every code word contains exactly twice as many binary digits as the information sequence it represents) and that any concatenation of the variable length code words satisfies the (2,7) constraint (each 1 in every code word is followed by at least two 0s,

and no code word begins with more than four 0s or ends in more than three 0s). It can also be verified that every information sequence can be decomposed uniquely into a sequence of variable length strings in the left-hand column of the table. In addition, the variable length code words constitute a prefix-free code (no code word is a prefix of another code word), so that unique decoding of code sequences can be accomplished. One can also describe a sliding-block decoder for this code in such a way that decoding errors due to a single code symbol in error cannot affect more than four user bits.

Prior to a few years ago, no general theory existed for the design of modulation codes (such as (d,k) codes) with minimum code word length, finite-state encoders, and sliding-block decoders, at rates arbitrarily close to capacity (or equal to capacity, when the capacity is a rational number). Now, however, there is systematic technique for code construction [8, 14]. The method, called the sliding-block code algorithm, allows for the design of a practical, efficient (d,k) code for any choice of the parameters d and k at any rational rate up to

User Data	Channel Bits
00	101
01	100
10	001
11	010

■ Table 4. Basic Encoding Table for (1,7) code

User Data	Channel Bits
00.00	101.000
00.01	100.000
10.00	001.000
10.01	010.000

■ Table 5. Substitution Table for violations in (1,7) basic encoder

Information Digits	Code Words
10	0100
11	1000
000	000100
010	100100
011	001000
0010	00100100
0011	00001000

■ Table 6. Encoding/decoding table for rate 1/2 (2,7) code

the capacity C , specifically for any integers m and n satisfying $m/n \leq C$, the algorithm yields a finite-state encoder that accepts m binary inputs and generates n binary outputs, and a state-independent decoder requiring only finite look-ahead and look-back, thereby limiting error propagation.

The sliding-block algorithm was first used to find a five-state encoder for a (1,7) code. It converts two binary information digits into three coded binary digits [12], and has been shown to generate the same set of code sequences as the (1,7) code described earlier. More recently, another rate 2/3 encoder with only four states [13], the minimum possible for a rate 2/3, (1,7) code [15],

The sliding block algorithm was first used to find a five-state encoder for a (1,7) code.

The nature of sliding-block codes is to propagate errors at the decoder input into a finite burst of decoded data errors.

was invented using the sliding-block code algorithm.

The nature of sliding-block codes is to propagate errors at the decoder input into a finite burst of decoded data errors. Therefore, random-burst correcting ECC have been applied as a sort of add-on to handle these rare, short burst errors. Single burst error correction codes and some specially crafted codes similar to Fire codes were used in some products, for example. Now, this design philosophy is slowly changing as engineers come to realize that error correction codes must be an integral part of any storage system and can lead to both higher reliability and storage efficiency. Today, the most general class of codes used are Reed-Solomon codes, although most products using Reed-Solomon codes are still not very aggressive in their error correction (e.g., the use of two-byte error correcting codes). Recently, however, several general purpose chips have become available [16, 17] that can correct (on the fly) many byte errors (e.g., 10) in a code word at speeds in excess of what is required for today's products. The optimization of the recording system, incorporating advanced modulation, detection, recording codes, and ECC is a challenging problem, beyond the scope of this paper, but it seems quite reasonable to conjecture that the newly available ECC power will find its way into future storage devices.

Partial Response Systems

Recently, researchers from IBM laboratories reported the results of an experiment demonstrating that an areal density of 1 gigabit per square inch could be achieved for the storage and reliable retrieval of digital data on a hard disk system [18]. This many-fold increase in density was achieved using a number of advanced techniques. One of these techniques was a different approach to combatting intersymbol interference, sometimes referred to as PRML, using partial-response (PR) signaling with maximum-likelihood (ML) sequence detection.

Instead of keeping the transitions far apart using (d,k) codes, PRML allows the transitions to be close together, and the read signal, with its resulting intersymbol interference, is equalized to a frequency response known as a class-4 partial-response channel [19]. The equalized signal is then detected by a maximum likelihood sequence estimator, i.e., a Viterbi detector [20]. In this section we will give a brief summary of the PR and ML components of this system as they apply to magnetic recording.

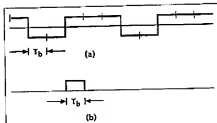
Partial Response Equalization

We now turn to a more detailed description of partial-response signaling. Consider the two-level write current, $I(t)$, shown in Fig. 16(a), where it should be noticed that the bit cell boundaries are now located such that the transitions occur at the edge of a bit cell. Shown in Fig. 16(b) is the elementary pulse $p(t)$.

Note that the write current $I(t)$ can be written in terms of the elementary pulse $p(t)$ as:

$$I(t) = \sum_{i=0}^{\infty} a_i p(t - iT_b)$$

where a_i takes on the values $+A$ or $-A$. Suppose that the channel is equalized so that the response



■ Figure 16. (a) Write current, (b) Elementary pulse $p(t)$

to the pulse $p(t)$ is a signal $h(t)$, referred to as the partial response signal. Then the readback voltage, $V(t)$, is of the form:

$$V(t) = \sum_{i=0}^{\infty} a_i h(t - iT_b)$$

Furthermore, assume that when $h(t)$ is sampled at bit interval boundaries every T_b seconds, the only nonzero samples are: $h_0 = h(0)$, $h_1 = h(T_b)$, \dots , $h_L = h(LT_b)$. The nonzero samples can be conveniently represented by the "partial-response polynomial" $h(D) = h_0 + h_1 D + \dots + h_L D^L$, where the factor D signifies a delay of t time units T_b . (In other words, $h(D)$ is the D -transform of the sampled pulse response.) The j th sample of the readback voltage $V(jT_b)$ can be written as:

$$V(jT_b) = \sum_{i=j-L}^j a_i h(jT_b - iT_b)$$

or

$$V(jT_b) = \sum_{i=j-L}^j a_i h_{j-i}$$

For magnetic recording systems with PW_{50}/T_b approximately equal to 2, comparatively little equalization is required to force the equalized channel to match a class-4 partial-response (PR4) channel where $h(D) = 1 - D^2$ [21]. At higher recording densities, one can choose a partial-response polynomial of the form $h(D) = (1 - D)(1 - D)^n$ where n is chosen as a positive integer greater than 1, to produce a response $h(t)$ that is a better match to the channel pulse response [22]. For PW_{50}/T_b approximately equal to 2.25, the proper choice of n is 2, leading to the so-called EPR4 (i.e., extended class-4 partial-response) channel with $h(D) = 1 - D - D^2 - D^3$ [22]. Eye diagrams for PR4 and EPR4 waveforms are shown in Figs. 17 and 18, respectively. These diagrams [19] represent the overlaying of the channel output signal seen in each time interval T_b , assuming a random binary input sequence. One can clearly see the nominal three-level (respectively, five-level) set of values at the sample times for the PR4 (respectively, EPR4) response. The eye diagrams provide some useful, qualitative indication of the robustness of the sample values at bit cell boundaries in the presence of additive noise and timing jitter.

Although PR equalization in communications systems and, as will be discussed, in magnetic recording systems is primarily used in conjunction with detection schemes that process samples of the channel output waveform, it is interesting to note that

it has also found application in recording systems employing peak detection. For example, raised-cosine and cosine-squared filters (corresponding to PR polynomials $h(D) = 1 + D$ and $h(D) = (1 + D)^2$) have long been used to achieve high frequency noise reduction and some degree of pulse eliminating, although the realizations have typically not been minimum-bandwidth. In addition, more recently it has been demonstrated that equalizing the channel to an extended PR with polynomial of the form $h(D) = (1 - D)(1 + D)^k$ provides performance improvement in peak detection systems [23, 24].

In a later section, we will describe a newly proposed coding method designed for a peak detection system employing EPR4 equalization.

PRML

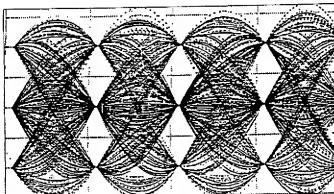
The applicability of partial-response signaling as a means of coping with intersymbol-interference in magnetic recording channels was suggested over 20 years ago [21], and the use of a Viterbi detector for maximum-likelihood sequence estimation in the storage context was proposed almost simultaneously with similar proposals for data communications [25, 26].

During the past 10 years, additional analysis, simulation and, finally, laboratory experimentation confirmed the potential value of the PRML system, [27-30]. In both simulation and experiments, the benefits in linear density that can be obtained over systems using RLL-coded peak detection have been found to be approximately 30 percent [31]. In addition, further research results indicated that the digital nature of the signal processing in PRML leads to advantages in electronic implementation (particularly VLSI) and extensibility, e.g., via coded-modulation [32, 33], digital adaptive equalization [34], and digital timing and gain control [30].

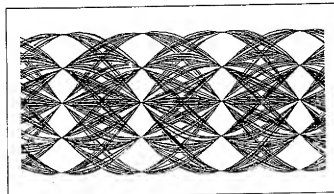
This activity has finally culminated in the incorporation of PRML into magnetic tape products, [35] and, very recently, magnetic disk products, [29].

We now briefly review the principles of Viterbi detection for combatting intersymbol interference, particularly as these concepts relate to coding for recording channels based upon partial-response. Recall that, in the NRZI ($1/(1 \oplus D)$) precoded channel, a code symbol 1 produces a positive or negative transition in the write current (respectively, a positive or negative pulse at the channel output) and a code symbol 0 produces no transition (respectively, pulse). This correspondence translated the restrictions on minimum and maximum transition spacing (respectively, run lengths of time intervals with no pulses at the input to the peak detector) into the easily represented (d, k) constraints.

Similarly, in the PRML setting, an Interleaved NRZI or INRZI ($1/(1 \oplus D^2)$) precoder converts the constraints on the samples at the input to the detector into simply described constraints on the code sequences applied to the input of the precoder, as we will describe. In the INRZI-precoded PR4 channel, a code symbol 0 at the input to the precoder will produce a sample 0 at the channel output, and a code symbol 1 will produce a sample value of either +1 or -1. The maximum-likelihood sequence detector based upon the Viterbi algorithm takes



■ Figure 17. Eye diagram for PR4 channel



■ Figure 18. Eye diagram for EPR4 channel

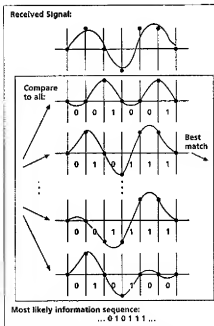
a received (noisy) sample sequence of length n ,

$$y = y_1 y_2 \dots y_n$$

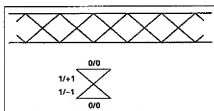
and determines in a recursive manner the channel output sequence (and the corresponding data sequence) that provides the best fit in a least-squares sense to the observed sample sequence, as shown schematically in Fig. 19 for the INRZI-precoded PR4 channel. (The authors wish to acknowledge that this visualization of ML detection is due to Gottfried Ungerboeck). A rough analogy to linear regression can be made, where one fits a line to a set of observations in such a way as to minimize the sum of squared errors.

The PR4 channel, with polynomial $h(D) = 1 - D^2$, can be considered as two time-interleaved "dicode" partial-response channels, each with polynomial $h(D) = 1 - D$. A common method of applying the Viterbi algorithm to maximum-likelihood sequence detection for the PR4 channel is to de-interleave the samples of the received data waveform to form two streams of samples, a stream made up of the samples at odd time indices and a stream made up of the samples at even time indices. Then two Viterbi detectors matched to the $1 - D$ channel can be used, one for each stream. In practice, one might even use just one detector in a pipelined fashion. This interleaving approach will be important when we come to discussing codes for this channel.

For an INRZI-precoded $1 - D$ channel, the



■ Figure 19. Maximum-likelihood detection schematic



■ Figure 20. Trellis representation of 1-D channel outputs

■ Figure 21. Difference-metric algorithm for 1-D Viterbi detector

Extension	Condition	Update
	$DM_k \leq 2y_{k+1} - 1$	$DM_{k+1} = 2y_{k+1} - 1$
	$2y_{k+1} - 1 < DM_k < 2y_{k+1} + 1$	$DM_{k+1} = DM_k$
	$2y_{k+1} + 1 \leq DM_k$	$DM_{k+1} = 2y_{k+1} + 1$

channel input/output sequences are conveniently represented by the trellis structure in Fig. 20.

Within each stage of the trellis, shown in the lower portion of Fig. 20, each edge is labeled with an input bit (before the slash) and a channel output symbol (after the slash). Input/output sequence pairs are generated by reading off the edge labels as one follows a path through the trellis from left to right. The two states at each time correspond to the parity of the number of input 1s so far, which determines the polarity of the channel output resulting from the next input 1. Alternatively, the states can be thought of as the polarity of the write current at the end of the last bit cell.

The Viterbi algorithm fits the time-indexed observations at the channel output with two allowable channel output sequences: one minimizing the sum of squared errors over all possible noiseless output sequences ending in the first state of the trellis at time n ; the other minimizing the sum of squared errors over the noiseless output sequences ending in the second state of the trellis at time n .

The recursive algorithm for finding these two "survivor" sequences is based upon a concept from dynamic programming called the principle of optimality. What it says, and what can be readily checked, is that each of the two survivor sequences at any time $n + 1$ must correspond to an extension of one of the survivor sequences from time n . The application of this principle to the 1-D channel leads to several elegant interpretations and implementations of the decoding algorithm. One description is based on the "difference metric," denoted DM_n , which is simply the difference of the accumulated squared errors for the two survivor sequences at a specified time n . The recursive "difference metric algorithm," first published in [36] (see also [30]), is described in Fig. 21. Note that only three of the four possible survivor extensions can occur.

This algorithm also can be interpreted as a "dynamic thresholding" scheme [37], as follows. Two threshold values, an upper threshold T_k^u and a lower threshold T_k^l , are initialized at time $n = 0$ to $+1/2$ and $-1/2$. If the sample value at time $n + 1$, y_{n+1} , falls in the upper interval, $y_{n+1} > T_k^u$, the first extension is selected, and the new thresholds are set to $T_{k+1}^u = y_{n+1}$ and $T_{k+1}^l = y_{n+1} - 1$. If y_{n+1} falls in the middle interval, $T_k^l \leq y_{n+1} \leq T_k^u$, the second (parallel) extension is selected, and the thresholds remain unchanged, $T_{k+1}^u = T_k^u$ and $T_{k+1}^l = T_k^l$. Finally, if y_{n+1} falls in the lower interval, $y_{n+1} < T_k^l$, the third extension is selected, and the thresholds are set to $T_{k+1}^u = y_{n+1} + 1$ and $T_{k+1}^l = y_{n+1}$. See Fig. 22.

The first and third extension possibilities cause the survivor sequences to "merge" and, therefore, in both survivor sequences, all decisions prior to the merge agree with those in the maximum-likelihood estimate that is ultimately generated. Note that, in the absence of noise, the merges take place precisely when the input bit is a 1.

The parallel extension option, on the other hand, defers the decision about the bit following the last merge until a future merge settles the matter. Consequently, the detector must keep a record of the survivors (called the path memory or trellis history) at least back to the last merge. In practical applications, it is therefore very desirable to force frequent merges by limiting the separation of 1s at the precoder input, thereby reducing the likelihood of

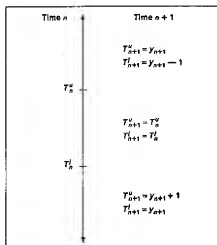


Figure 22. Dynamic threshold interpretation of Viterbi detection

errors caused by truncation of the path memory to a manageable length. As mentioned earlier, in recording systems the preferred practice is to guarantee such a property of the recorded sequences by means of constrained coding, rather than to achieve it only probabilistically via scrambling. This constraint on the input will play a role in the next section when we discuss the recording code constraints for PRML channels.

In general, if the partial-response polynomial is of degree L , and if the input to the channel is a binary sequence, then the Viterbi detector will require $2L$ states. For example, the Viterbi detector for the EPR4 partial-response channel, where $h(D) = 1 + D - D^2 + D^3$, uses an eight-state trellis, shown in Fig. 23. Here the trellis states represent the write-current levels, denoted 0 and 1, at the ends of the last three-bit cells.

An important performance indicator for a trellis-based detector is the free squared-Euclidean distance which, roughly speaking, measures the separation between the sequences most likely to be confused when corrupted by channel noise. The free distance can be derived from the trellis, as follows. We consider the sum of the squared differences between the noiseless outputs for every pair of paths in the trellis that start in a common state and end in a common (but perhaps different) state. The minimum of this sum is the free (squared-Euclidean) distance. For the $1-D$ trellis shown in Fig. 20, the free squared-Euclidean distance is equal to 2.

In a channel with additive white Gaussian noise, with zero mean and standard deviation σ , the probability of an error event, for moderate to high SNR, is then well-approximated by:

$$Pr(\text{error event}) = NQ\left(\frac{d_{\text{free}}}{2\sigma}\right)$$

where N is a constant determined by the trellis, and $Q(x)$ is the familiar complementary error function

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$$

We remark that sometimes in defining the free

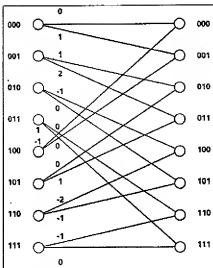


Figure 23. Eight-state trellis for EPR4 channel

squared-Euclidean distance we do not insist that the ending state be the same for the two paths. Of course, the expression for the probability of error event must then be modified accordingly. With this definition, for example, the free squared-Euclidean distance for the $1-D$ trellis would be equal to 1.

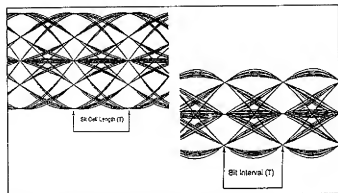
Codes for Partial Response Channels

In this section, we will describe several coding techniques developed for use in recording systems using partial response equalization. We will first describe a code designed for a new peak detection system incorporating EPR4 equalization. We then turn to a class of constrained codes, denoted $(0, G/I)$ codes, that have been used in the implementation of PRML channels in disk drives. These constraints incorporate the analogue of the k constraint in (d, k) codes, as well as a new constraint that affects the required length of the Viterbi detector path memory. Finally, we turn to the consideration of codes intended to improve the noise immunity of partial-response channels, and we indicate several exploratory directions proposed for trellis-coded modulation in storage channels.

Codes for Peak Detection for the EPR4 Channel

This section describes an exploratory technique for extending the use of peak detection to the EPR4 channel [38]. Recall that the EPR4 channel corresponds to the partial-response polynomial $h(D) = 1 + D - D^2 + D^3$. The eye pattern for this equalization (using the minimum bandwidth signal having these sampled values) was given in Fig. 18. It should be noted from this figure that the channel output waveforms exhibit a number of different types of peaks with varying amplitudes and that, at the normal sampling point for the EPR4 channel, the waveforms can assume five different values.

If one restricts the input waveform so that the cor-

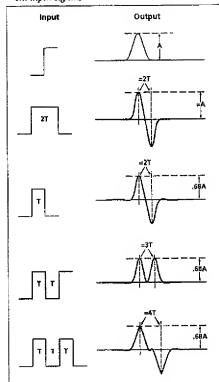


■ Figure 24. (a) Eye pattern for (1,7)-coded EPR4 Channel
(b) Eye pattern for INRZI-precoded (1,7) EPR4 Channel

responding binary sequence (using NRZI modulation) satisfies the (1,7) constraint, the resulting eye pattern is as shown in Figure 24(a).

Now all but the large peaks have been eliminated by the code, and a peak detector could be used to identify transitions in the input write signal. At the normal sampling point for the EPR4 channel the waveform still has five levels. If the (1,7) coded sequence is first passed through a precoder with transfer function $1/(1 \oplus D^2)$, the resulting eye pattern is as shown in Figure 24(b). Now only the smallest peaks are retained in this waveform, and once again a peak detector can be used to recover the input write current. At the normal sam-

■ Figure 25. Response of EPR4 channel to different input signals



pling point for the EPR4 waveform, this coded waveform has only three levels.

Figure 25 shows the response of the EPR4 channel to a number of different input waveforms. It is important to note that a pulse of duration $2T$ (here T is the sampling interval) results in a pair of peaks of heights $+.68A$ and $-.68A$ separated by approximately $2T$, while a pulse of duration T produces a pair of peaks of height approximately $+.68A$ and $-.68A$ and separated again by almost $2T$. If the number N of input transitions spaced T apart exceeds 2, the output waveform will contain two peaks of amplitude $.68A$ spaced approximately NT apart, as illustrated for $N = 3, 4$ at the bottom of Fig. 25. In general, the peaks of amplitude A are out of phase by $1/2$ a bit cell from the peaks of amplitude $.68A$. When the basic waveforms shown in Fig. 25 are concatenated with each other, the shapes of the output waveforms remain approximately as shown in Fig. 25, provided that the last transition of the preceding basic waveform is no closer than $3T$ from the first transition of the basic waveform that follows it. The basic waveforms producing large peaks need only be separated by two T or more.

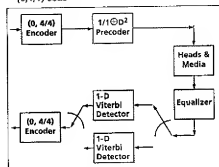
Using these observations as motivation, a code has been devised for use with this INRZI-precoded EPR4 channel (where the precoder has transfer function $1/(1 \oplus D^2)$) such that a peak detector that can reliably detect peaks and discriminate between large and small peaks can identify the binary sequence that corresponds to the write current. The code also produces every eight bit cells for timing recovery. The maximum possible rate for this code is $C = .8485$, which is approximately 25 percent higher than for a (1,7) code.

(0, GII) Codes for the PRML Channel

The gigabit per square inch experiment and the first United States disk drive product employing PR4 equalization utilized code sequences satisfying a new type of constraint called a (0, GII) constraint. This constraint imposes run length limitations that aid timing and gain recovery and simplify the design of the Viterbi detector for the channel. A block diagram of the PRML system using this code is shown in Fig. 26. The channel response is equalized to that of the PR4 system. As described previously, the PR4 Viterbi detector is decomposed into a pair of interleaved detectors matched to the 1-D channel.

The constraints on the PR4 channel outputs

■ Figure 26. Block diagram of system using a (0, GII) code



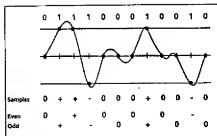


Figure 27. Waveform and samples for (0,4/4) sequence

are twofold. First, for accurate timing and gain recovery, it is desirable to limit the number of consecutive zero samples in the noiseless PR4 channel output sequences generated by the code sequences. Second, in order to minimize performance degradation resulting from the truncation of the path memory in the interleaved $1-D$ Viterbi detectors, the maximum run length of zero samples in each of the two interleaves comprising a channel output sequence should be limited.

As mentioned previously, the INRZI-precoded PR4 channel generates a sample 0 at the channel output for a code symbol 0 at the input, and a sample value of either +1 or -1 at the output for a code symbol 1 at the input. Therefore, the first constraint translates into a global constraint, denoted by the symbol G , on the maximum run length of 0 symbols in any code string. This constraint is essentially the same as the k constraint in the (d, k) codes. The second constraint corresponds to an interleaved constraint, denoted by l , on the maximum run lengths of 0 symbols in each of the interleaves of a code sequence. The "0" in the notation $(0, G/l)$ can be treated as the analogue of the global con-

$(0, G/l)$	Capacity	Code Rate	Encoder States	Decoder Window
$(0, 4/4)$.961	8/9	1	1
$(0, 4/3)$.939	8/9	3	1
$(0, 3/6)$.944	8/9	1	1
$(0, 3/5)$.942	8/9	2	1
$(0, 3/4)$.934	8/9	3	2
$(0, 3/3)$.915	8/9	4	2

Table 7. Parameters of some rate 8/9 $(0, G/l)$ codes

straint. Here, it serves to emphasize that there is no restriction on the minimum separation of nonzero samples at the channel output; that is, the code sequences are not forbidden to have adjacent 1s. Figure 27 shows an example of a code sequence satisfying a $(0, 4/4)$ constraint, similar to the constraints used in the commercial PRML channel, along with the corresponding noiseless channel output waveform, the global sample sequence, and the interleaved sample sequences. Note that adjacent nonzero samples can have the same polarity, that is, they do not correspond directly to peaks in the output waveform.

As with the (d, k) constraints, there are simple finite-state transition diagrams that describe the $(0, G/l)$ constraints from which one can compute capacities and construct sliding-block codes. The capacities of several $(0, G/l)$ constraints of practical interest, as well as the parameters of rate 8/9 codes satisfying these constraints [39] are given in Table 7.

J. Eggenberger first discovered the optimal block list of length 9 for the $(0, 4/4)$ and $(0, 3/6)$ constraints, i.e., the largest collection of nine-bit code words that satisfy the prescribed constraints when freely concatenated. The $(0, 4/4)$ optimal block list contains 279 words, listed in decimal representation in Table 8.

Table 8. Maximal list of length-9 words for the $(0, 4/4)$ constraint

73	116	183	225	268	310	361	402	438	479
75	117	185	227	269	311	363	403	439	481
76	118	186	228	270	312	364	406	441	483
77	119	187	229	271	314	365	407	442	484
78	121	188	230	281	315	366	409	443	485
79	122	189	231	282	316	367	410	444	486
89	123	190	233	283	317	369	411	445	487
90	124	191	235	284	318	370	412	446	489
91	125	195	236	285	319	371	413	447	491
92	126	198	237	286	329	372	414	451	492
93	127	199	238	287	331	373	415	454	493
94	146	201	239	289	332	374	417	455	494
95	147	203	241	291	333	375	419	457	495
97	150	204	242	292	334	377	420	459	497
99	151	205	243	293	335	378	421	460	498
100	153	206	244	294	345	379	422	461	499
101	154	207	245	295	346	380	423	462	500
102	155	210	246	297	347	381	425	463	501
103	156	211	247	299	348	382	427	466	502
105	157	214	249	300	349	383	428	467	503
107	158	215	250	301	350	390	429	470	505
108	159	217	251	302	351	391	430	471	506
109	177	218	252	303	353	393	431	473	507
110	178	219	253	305	355	395	433	474	508
111	179	220	254	306	356	396	434	475	509
113	180	221	255	307	357	397	435	476	510
114	181	222	265	308	358	398	436	477	511
115	182	223	267	309	335	399	437	478	

For the (0, 3/6) constraint, Eggenberger found two optimal block lists containing 272 code words; the code words in one code being the time-reverse of the code words in the other. For these constraints, specific rate 8/9 codes were obtained by selecting 256 code words from these optimal lists. (For more details about these codes and their logic implementations, see [8, 39, 40].)

Codes for Improving Noise Immunity for the PR4 Channel

The main reason for choosing the particular class of partial-response channels discussed previously is that the required equalizers do not boost the noise power excessively. The equalizer does change the spectral characteristics of the noise somewhat, but often this factor is ignored in the design and analysis of the Viterbi detectors (when the slight loss in accuracy and performance is deemed acceptable).

As mentioned previously, the performance of the Viterbi detector at high signal-to-noise ratios (SNR) is known to be governed by the free squared-Euclidean distance of the code, and the free squared-Euclidean distance for the 1-D channel is equal to 2. Codes for improving the noise immunity for the PR4 channel involve eliminating some of the sequences generated by paths through the trellis shown in Fig. 28, with the objective of increasing the free squared-Euclidean distance by a coding gain factor g , where $g > 1$. In any coded system of this kind, one is trading information rate (represented by the code rate R) and complexity (of the encoder/decoder logic as well as the modified detector) for coding gain. The parameter Rg is called the asymptotic coding gain (ACG) for the coded system, and to a first approximation represents how much more (or less) noise can be tolerated by the coded system as compared with an uncoded system that yields the same error performance (at low error rates).

The interleaved-FM code for the PR4 channel provides a simple, but representative, example of this tradeoff and the potential value of similar coded-modulation schemes. The channel configuration is the same as in the previous section, with the PR4 channel decomposed into two interleaved 1-D channels, each preceded by a 1/(1-D) precoder. The interleaved-FM code, as the name suggests, is defined by applying separately to the even and odd interleaves of the data string a simple rate 1/2 code, often referred to as the FM (frequency modulation) code. The FM code is a block code with the encoding rule:

$$\begin{aligned} 0 &\rightarrow 01 \\ 1 &\rightarrow 11 \end{aligned}$$

The interleaved-FM code therefore satisfies a (0, G/1) = (0, 2/1) constraint. We remark that the sequences produced at the precoder output comprise the widely known biphasic code, a simple rate 1/2 block code with code words 01 and 10. The precoded PR4 channel with the interleaved-FM code is therefore equivalent to the (unprecoded) PR4 channel with interleaved-biphase coding, and we will use the names interchangeably. The trellis describing the output sequences as a function of the input data in the FM-coded 1-D channel is shown in Fig. 28. If we use a pair of detectors in an interleaved fashion for detection of the interleaved-FM coded PR4 channel, as was done for

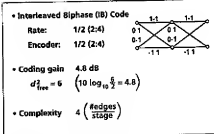


Figure 28. Trellis structure for FM-coded 1-D channel

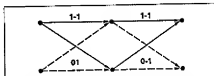


Figure 29. Minimum distance event for FM-coded 1-D channel

the uncoded PR4 channel, it suffices to examine this trellis. The minimum squared-Euclidean distance generated by an error event is 6, so the distance gain factor is $g = 3$. An example of such a minimum distance event is depicted in Fig. 29. The ACG is therefore given by:

$$\text{ACG} = 10 \log_{10} Rg = 10 \log_{10} \frac{1}{2} \cdot 3 = 1.8 \text{ dB}$$

To assess the added hardware complexity associated with the trellis-coded scheme, one can use rough meaningful measures for the encoder/decoder logic and Viterbi detector. Specifically, for the encoder/decoder, one might look at the number of states and the data and code word length in the finite-state encoder, along with the length of the sliding-block decoder window. With regard to this measure, the FM code adds only minor complexity relative to the uncoded channel: the encoder has only one state and can be implemented by simply inserting a 1 following each data bit; the decoder window is two code bits long, and decoding is implemented by simply dropping the second code bit of each detected pair.

From the trellis structure, we can get a qualitative indication of the hardware complexity by looking at the number of states (each corresponding to an Add-Compare-Select processor), the number of edges per trellis stage (the number of possible survivor extensions that need to be examined), and the number of samples detected per trellis stage. In the FM case, there are only two states, four edges per stage, and two samples per stage, again representing a minimal increment in complexity relative to the uncoded PRML detector. There is even a simple difference metric form for the Viterbi algorithm [41].

As an approach to increasing linear density, the advantage of such a coded-PR4 system, although possibly substantial, may not be apparent without careful analysis of the channel signal-to-noise ratio as a function of transition density and the required channel equalization [42]. In disk recording systems, however, there is a second coordinate, corresponding to the radial direction, along

which density can be increased. Overall areal density can be optimized by choosing the appropriate balance of linear and radial (i.e., track) density. In this setting, the potential benefit of coded systems can be appreciated by considering the following interesting, although simplistic, argument.

Suppose one begins with a benchmark PRML channel using a rate $8/9$ (0,4/4) code on a nominal track width of W , with linear density L . Let's assume that head and servo technology permit the physical reduction of track width by a factor of three. One could then divide the original track into three subtracks, each of width $W/3$. Theory and experiment [43] indicate that the amplitude of the readback signal and the noise power due to the medium would both be reduced by approximately this factor of three, corresponding to a 4.8 dB signal-to-noise ratio penalty. If, on each subtrack, the original channel scheme were applied (leading to a tripling of the areal density) the performance would be unacceptably poor due to the SNR loss. However, if we apply the rate $1/2$ interleaved-FM code on each subtrack, the gain factor g exactly offsets the SNR loss, implying that the probability of error on each subtrack will be virtually unchanged from the nominal value. The catch, of course, is that the code rate on each subtrack is now $1/2$, reducing the linear density per subtrack to

$$\left(\frac{1}{2}\right)L = \frac{9}{16}L.$$

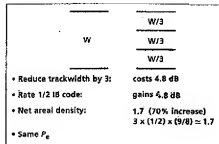
The track density has been tripled, however, leading to an overall areal density that is a factor of $27/16$ (or approximately 1.7) times the original, implying a 70 percent increase (Fig. 30).

Although this back-of-the-envelope calculation ignores several important technology issues, such as intertrack interference, narrow-track width head design, and position-servo accuracy, it at least suggests that the development of practical coded systems might provide a route to significant increases in areal density in disk drives.

The example clearly illustrates that the objective is to design codes with high rate and large coding gain, in order to minimize the track width reduction and to provide the greatest noise immunity and areal density increase in the scenario just sketched. Soon we will give examples of codes with rate $4/5$ and gain factor $= 2$. A calculation similar to above shows that cutting the track width by two and applying a code with these parameters on each subtrack provides an estimated areal density increase of almost 80 percent.

Several methods have been found for the

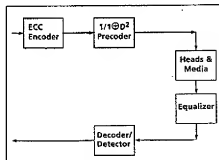
■ Figure 30. Track cutting for increased areal density



design of coded systems with ACG greater than 1, for a range of code rates. The simplest such scheme is to use an ordinary binary ECC code and a precoder [32] as shown in Fig. 31. (See also [44].) Assume that the ECC code has rate R and minimum Hamming distance equal to d_{min}^H , so that when the code is used as an error correction code over a random binary channel it is capable of correcting $(d_{min}^H - 1)/2$ or fewer errors. Recall that an input 0 to the precoded channel produces an output 0 and an input 1 produces an output +1 or -1. It follows that the minimal squared-Euclidean distance will satisfy the inequality

$$d_{min}^2 \geq d_{min}^H.$$

In fact, it can be shown that for the coded PR4



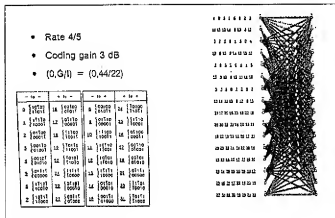
■ Figure 31. Block diagram of coded system for noise immunity

system incorporating this code d_{min}^H must be even. Therefore, the coded channel has an ACG bounded below by $Rd_{min}^H/2$ if d_{min}^H is even or $R(d_{min}^H + 1)/2$ if d_{min}^H is odd.

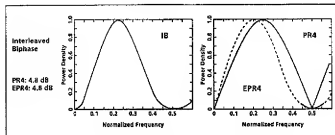
To get the full benefit of any coding gain, the decoder for this system should be matched both to the code and also to the precoded $1 - D^2$ partial-response channel. Just as in the uncoded case, one could bit-wise interleave code words and treat the channel as two $1 - D$ channels. If the ECC is a convolutional code, the combination of the code and the $1 - D$ channel can be decoded by a single trellis. If one interleaves a convolutional code with a 2^k state encoder, the resulting combined Viterbi decoder/decoder operates on a trellis with at most 2^{k+1} states.

The coding gain bound suggests that for a given rate, one would like to use an ECC with the largest minimum Hamming distance. Optimal convolutional codes have been found for a wide range of rates and trellis complexity by computer search, and tables of these are now available in many textbooks [45]. Using these codes, the lower bound on minimum distance of the trellis-coded PR4 channel has been found to be tight in virtually all cases.

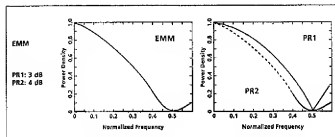
One still has to limit the global runs of consecutive 0s entering the precoder in order to guarantee a usable signal for timing recovery, as well as the runs of 0s on the interleaves to reduce path memory requirements. One approach to accomplishing this is to use a coset of the ECC code, i.e., an additive translation of the code sequences, obtained by the componentwise addition (modulo 2) of a fixed binary sequence to each code sequence [32]. For a rate k/n code, one could do this by complementing the code symbols in a specific set of positions



■ Figure 32. Trellis structure for rate 4/5 code with ACG 2.0 dB



■ Figure 33. Power spectrum of Interleaved-FM and PR4/EPR4 frequency response



■ Figure 34. Power spectrum of EMM and PR1/PR2 frequency response

in each code word of length n . (Another approach might be to first encode the data using a $(0, k)$ code, and then to apply a systematic error correcting code with the desired Hamming distance.)

An example of the precoded ECC scheme, given in [32], makes use of a coset of a rate 4/5 convolutional code, with minimum Hamming distance 3, applied to each of the interleaved 1-D channels comprising the PR4 channel. The resulting gain factor is $g = 2$, providing ACG equal to 2 dB. The decoding trellis for each interleaved, derived from the eight-state trellis of the underlying convolutional code, has 16 states, as shown in Fig. 32 [32, 33]. This code satisfies the run length constraints $(0, G/I) = (0, 44/22)$.

The reader may have observed that the interleaved-FM code, which has a minimum Hamming distance 1 and is not particularly attractive as an ECC code, achieves a coding gain that far exceeds the

lower bound just derived. This code is an example of another class of codes, called matched-spectral-null (MSN) codes, which recently have been shown to provide an efficient method of achieving moderate coding gain at high rates. An MSN code is designed in such a way that the average power spectrum of the write-current waveforms generated by the code has the value zero (i.e., a spectral null) at frequencies where the partial-response channel frequency response is zero.

Aside from the relative reasonableness of this design criterion (why waste signal power by transmitting at frequencies where the channel response is zero?) it has been shown mathematically that matching of the code and channel spectral null frequencies provides significant coding gain for a large class of partial response channels, including those relevant to magnetic and optical recording channels [33].

For example, Fig. 33 shows the power spectrum of the interleaved-biphase code or, equivalently, the precoded interleaved-FM code, alongside the frequency response of the PR4 channel, as well as the frequency response of the EPR4 channel, where it also provides ACG of 1.8 dB.

Another example of an MSN code, intended for optical recording channels is the rate 2/3, even-mark modulation (EMM) code [46]. It has a spectral null at the Nyquist frequency (one-half the recorded symbol frequency) and provides coding gain factors $g = 2$ and $g = 2.5$ for the 1 + D (class 1 or PR1) and $(1 + D)^2$ (class 2 or PR2) partial-response channels. The corresponding power spectral density and frequency response curves are shown in Fig. 34.

Returning to the 1-D channel, it has been shown [33] that the minimum squared-Euclidean distance at the output of the coded channel is bounded below by $2K$, where K is the order of the code's spectral null at zero frequency (meaning that the first $2K - 1$ derivatives of the code power spectrum are zero at zero frequency).

Practical codes with spectral nulls of a given order at specified frequencies can be designed using the sliding-block code construction techniques alluded to earlier. We remark that the spectral null constraints automatically provide the necessary run length constraints characteristic of the $(0, G/I)$ codes described in the previous section, as illustrated by the interleaved-biphase code. The initial FSTD used in the code design procedure is chosen to describe a family of spectral null sequences with capacity large enough to satisfy the target code rate. For example, Fig. 35 shows such a so-called canonical diagram from which one can extract initial FSTDs describing sequences with a spectral null at zero frequency [33]. (The labels should be interpreted as write-current levels.)

A rate 8/10 spectral null code with a four-state encoder, satisfying $(0, G/I) = (0, 10/5)$ constraints when interleaved, was designed for the 1-D channel. Unfortunately, as might be expected, the complexity of the encoder finite state machine generally increases as the code rate does, so the trellis structure reflecting the combination of the modulation code and channel can be quite complicated for high code rates. Indeed, in the case of the rate 8/10 MSN code, the trellis would have eight states, and 256 branches emanating from each state.

As shown in [33], however, there is a natural, reduced-complexity Viterbi detector for MSN codes that asymptotically achieves maximum-likelihood performance as a function of the signal-to-noise ratio. The detector structure is based on the much simpler trellis derived from the initial FSTD used in the sliding-block code construction. The MSN code sequences belong to the supercode of spectral null sequences that are represented by the reduced-complexity trellis, and the MSN code can be designed to ensure that, for any code sequence, no sequence generated by the trellis is closer to the code sequence (in Euclidean distance) than the minimum Euclidean distance of the code. Thus, the decoder can apply the Viterbi algorithm to the reduced-complexity trellis to find the spectral null sequence in the supercode that best fits the noisy channel output sequence. In the unlikely event that the sequence produced by the detector is in error or is not in the range of the MSN code, the sliding-block decoder limits the propagation of errors in the decoded data sequence. For the rate 8/10 code, the corresponding reduced-complexity trellis is shown in Fig. 36. Issues related to the VLSI implementation of an exploratory rate 8/10 MSN code for PR4 are discussed in more detail in [47].

A recently proposed alternative method of achieving coding gain is shown in Fig. 37. Again, a code with good Hamming distance is used for the ECC code. Now, however, we require that the code be chosen from the class of codes that allow for efficient decoding via a soft decision decoding algorithm. (All convolutional codes fit into this class.) A two-step detector is now used. The first step uses an enhanced Viterbi detector matched to the channel itself, but modified so that the detector outputs reliability estimates on the binary data in the detected data stream. The second step invokes a soft decision decoder for the ECC code that uses the output of the first Viterbi detector as its input. This concatenated detector approach, using an extended BCH code with code word length 64 and minimum Hamming distance six, has been shown to achieve an asymptotic coding gain of close to 3 dB [48].

Other Techniques

Several other promising techniques have been proposed to increase the density of recording. One of these, familiar to communication engineers, is decision feedback equalization [49, 50]. Another is a limited tree search algorithm [51]. There also is an interesting hybrid technique, using portions of the peak detection system and the PRML approach to detect (1,7)-coded information [52]. These topics have not been treated here, but the reader is encouraged to consult the references for details.

Summary

In this paper, we have discussed many of the types of modulation codes designed for use in storage devices using magnetic recording. The codes are intended to minimize the negative effects of intersymbol interference. Most commercial disk drives today use a simple type of detector called a peak detector, and a corresponding class of codes called run

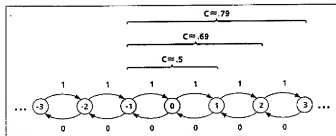


Figure 35. Canonical diagram for spectral null at zero frequency

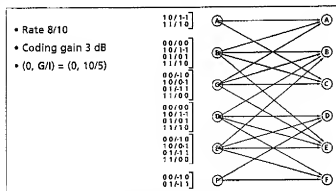


Figure 36. Trellis structure for rate 8/10 MSN code

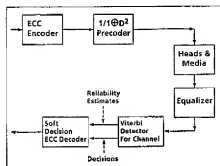


Figure 37. A pair of decoders that use soft decisions

length-limited (d, k) codes have found wide application. Recently, another recording channel technology, based on sampling detection-partial-response (or PRML), has been introduced in commercial disk drives. This technology hinges on the use of controlled intersymbol interference, and it requires a new class of codes, called (0, G/I) codes.

The paper concluded with several examples illustrating that the introduction of partial response equalization, sampling detection, and digital signal processing has set the stage for the invention and application of advanced modulation and coding techniques in future storage products.

Acknowledgment

The authors are grateful to our many colleagues, particularly those at the Center for Magnetic Recording Research (CMRR) and IBM, who work in the field of signal processing and coding, and whose technical

results form the basis for this article. We also gratefully acknowledge support from National Science Foundation Grant DMB-8920428, the Center for Magnetic Recording Research (UCSD), and the IBM Corporation.

References

- [1] J. C. McMillan, "A Unified View of High Density Digital Recording Theory," *IEEE Trans. Magn.*, vol. MAG-18, no. 3, pp. 1166-69, Sept. 1982.
- [2] P. Siegel, "Applications of a Peak Detection Channel Model," *IEEE Trans. Magn.*, vol. MAG-18, no. 3, pp. 1250-52, Nov. 1984.
- [3] L. Barbaresi, "Minimum Noise Pulse Shlimmer," *IEEE Trans. Magn.*, vol. MAG-17, no. 6, pp. 1349-52, Nov. 1981.
- [4] E. Bielekamp, "The Technology of Error-Correcting Codes," *Proc. IEEE*, vol. 68, no. 3, pp. 566-59, May 1980.
- [5] P. Siegel, "Recording Codes for Digital Magnetic Storage," *IEEE Trans. Magn.*, vol. MAG-21, no. 3, pp. 1344-49, Sept. 1985.
- [6] K. A. Schaubman, "Convolutional Coding Techniques for Digital Recording," *IEEE Trans. Magn.*, vol. 21, no. 3, pp. 1349-59, Nov. 1980.
- [7] K. A. Schaubman, "Convolutional Coding Techniques for Digital Recording," (Printed on film), 1981.
- [8] R. Marcus, P. Siegel, and J. Wolf, "A Tutorial on Finite-State Modulation Codes for Data Storage," *IEEE J. Select. Areas Commun.*, vol. 3, pp. 331-45, Sept. 1985.
- [9] R. Marcus, "Signal Processing and Coding for Recording Channels: To appear."
- [10] F. J. van den Broek, "The Complete Handbook of Magnetic Recording," (TAE Books), 1980.
- [11] A. Patel, "Zero Modulation Encoding in Magnetic Recording," *IEEE Res. Dev.*, vol. 19, no. 4, pp. 366-74, July 1975.
- [12] G. Jacoby, "A New Look-Ahead Code for Increased Data Density," *IEEE Trans. Magn.*, vol. MAG-11, no. 5, pp. 1002-04, Sept. 1975.
- [13] R. Adler, M. Havner, and J. Mousouris, "Method and Apparatus for Generating a Noiseless Shlimmer-Bell Code for a (1,7) Channel with Rate 1/2," U.S. Patent 4,412,825, 1982.
- [14] A. Weathers and J. Wolf, "A New Rate 2/3 Shlimmer-Bell Code for the (1,7) Runlength Constraint with the Minimal Number of Encoder States," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pt. II, pp. 908-13, May 1991.
- [15] R. Adler, D. Cooperman, and M. Hassner, "Algorithms for Shlimmer-Bell Codes," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 5-22, Jan. 1993.
- [16] S. Marcus and R. Roth, "Bounds on the Number of States in Encoder Design for Input Constrained Channels," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pt. II, pp. 1028-32, May 1991.
- [17] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [18] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [19] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [20] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [21] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [22] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [23] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [24] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [25] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [26] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [27] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [28] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [29] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [30] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [31] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [32] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [33] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [34] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [35] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.

- [36] Channel, Proc. of the Fifth International Conf. On Video and Data Recording, Hampshire, England, pp. 151-57, April 1984.
- [37] M. M. Ferguson, "Optimal Reception for Binary Partial-Response Channels," *Bell Syst. Tech. J.*, vol. 51, no. 4, pp. 493-505, Feb. 1972.
- [38] R. Wood, "Owner's Manual," *IEEE Spectrum*, vol. 27, no. 5, pp. 32-39, May 1990.
- [39] A. Viterbi and J. Wolf, "Performance Evaluation of the New Coding Scheme for the Peak Detecting Magnetic Recording Channel," *IEEE Trans. Magn.*, vol. 27, no. 11, November 1989.
- [40] A. Marcus and J. Wolf, "Convolutional Coding for the Magnetic Recording Channel," *IEEE Trans. Magn.*, vol. 27, no. 11, November 1989.
- [41] J. E. Biegel and A. A. Patel, "Method and Apparatus for Implementing Optimum PPM Codes," U.S. Patent 4,707,881, issued November 17, 1987.
- [42] J. E. Biegel, A. A. Patel, and P. Siegel, "Difference Metric Decoder for Interleaved Biphasic Trellis Codes," *IBM Technical Disclosure Bulletin*, vol. 31, no. 7, pp. 476-81, Dec. 1988.
- [43] K. A. Schaubman, "Convolutional Coding Techniques for the Magnetic Recording Channel," *IEEE Trans. Commun.*, vol. 37, no. 13-19, May 1989.
- [44] S. L. Lumsden, et al., "Reduction of Edge Noise in Thin Film Metal Media Using Discrete Tracks," *IEEE Trans. Magn.*, vol. 25, no. 3, pp. 331-35, Sept. 1989.
- [45] A. R. Calderbank, C. Hoegard, and T. A. Lee, "Binary Convolutional Codes with Application to the Magnetic Recording Channel," *IEEE Trans. Inf. Theory*, vol. 37, no. 5, pp. 797-815, Nov. 1988.
- [46] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, 1983.
- [47] R. Kozak and P. Siegel, "Even-Mark Modulation for Optical Recording," *Proc. of the 1988 Intl. Conference on Communications ICC-88*, vol. 3, pp. 1619-23, Boston, MA, June 1988.
- [48] C. E. Shannon, et al., "A Digital Magnetic Recording Channel," *Digest of the 1988 IEEE Symposium on Information Theory and Coding*, pp. 1-11, San Francisco, CA, Feb. 13-15, 1991.
- [49] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [50] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [51] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [52] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [53] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [54] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [55] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [56] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [57] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [58] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [59] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [60] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [61] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [62] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [63] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [64] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [65] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [66] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [67] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [68] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [69] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [70] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [71] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [72] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [73] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [74] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [75] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [76] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [77] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [78] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [79] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [80] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [81] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [82] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [83] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [84] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [85] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [86] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [87] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [88] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [89] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [90] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [91] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [92] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [93] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [94] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [95] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [96] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [97] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [98] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [99] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.
- [100] J. H. P. Probert, "Specification 4010, 'High Speed Read Solomonic Encoder/Decoder,'" IBM, 1980.

Biography

PAUL H. SIEGEL was born in Berkeley, California in 1953. He received the B.S. degree in Mathematics in 1975 and the Ph.D. degree in Mathematics in 1978, both from the Massachusetts Institute of Technology. He held a Chaim Weizmann fellowship during a year of postdoctoral study at the Courant Institute, New York University. He joined the research staff at IBM in 1980. He is currently manager of the Signal Processing and Coding project at the IBM Almaden Research Center in San Jose, California. His primary research interests are the mathematical foundations of signal processing and coding, especially as applicable to digital data storage channels. He holds several patents in the area of coding and detection for digital recording systems. He has taught courses in information and coding at the University of California, Santa Cruz and at Santa Clara University. He was a Visiting Associate Professor at the University of California, San Diego while at the Center for Magnetic Recording Research during the 1989-90 academic year. Dr. Siegel was elected to the Santa Clara Hall of Fame in 1974. He is a member of the IEEE, and is currently a member of the Board of Governors of the IEEE Information Theory Society. He was a Guest Editor of the May 1991 Special Issue on Coding for Storage Channels of the IEEE Transactions on Information Theory.

JACK KASZUBA received the B.S.E. degree from the University of Pennsylvania, Philadelphia, in 1956, and the M.S.E. and Ph.D. degrees from Princeton University, Princeton, in 1958 and 1963, respectively. He was a member of the Electrical Engineering Department at New York University from 1953 to 1965, and was the Polytechnic Institute of Brooklyn from 1965 to 1973. He was Chairman of the Department of Electrical and Computer Engineering at the University of Massachusetts from 1973 to 1975, and was Professor of Electrical Engineering at the University of Massachusetts from 1975 to 1980. He has been a Chief Professor of Electrical Engineering and Computer Engineering and a member of the Center for Magnetic Recording Research at the University of California, San Diego. He also held a part-time appointment at Qualcomm, Inc., San Diego, CA. His current interests are in signal processing for storage channels. He has been a member of the IEEE Information Theory Society from 1974. He was International Chairman of the IEEE Conference on Information Theory, San Francisco, 1980. He was a recipient of the 1980 E. H. Snodgrass Achievement Award from the IEEE Communications Society and was co-receptor of the 1975 IEEE Information Theory Group Paper Award for his paper "A New Coding Scheme for Correlated Information Sources" (coauthored with Dr. Siegel). From 1971 to 1972, he was an NSF Senior Postdoctoral Fellow, and from 1973 to 1980 he held a Guggenheim Fellowship.